HP-41C/41CV/41CX

Module de fonctions Gestion de tableaux

Manuel de référence

Fabricants			
Distributeurs			
Total 5			
	8,		
Fabricants	0,00		
Distributeurs	2,74		
Total 7	11,52	7	
Fabricants	0,00	0,0	
Distributeurs	0,00	0,00	
Total 10	0,00	0,00	0
	2,20	45,45	5
Fabricants	0,00	0,00	0
Distributeurs	2,64	54,55	4
Total 15	4,84	29,59	9
Total B	16,36	54,92	28

CONSEILLÉ		
Linéaire	% lin	- Références
,00	0,00	0
	0,00	0
	0,00	0
	66,00	13
	,00	0
	4,00	5
	57,01	18
0,00	0,00	0
0,00	0,00	0
0,00	0,00	0
2,46	45,00	6
0,00	0,00	0
3,01	55,00	5
5,47	42,99	11
12,71	56,10	29

PREFACE

Lorsqu'en 1979 apparut le HP-41, il représentait un étonnant saut qualitatif dans sa catégorie; il autorisait l'introduction, le traitement et l'affichage de chaînes de caractères. La voie était ouverte pour un dialogue complet entre le calculateur et son utilisateur, dialogue que favorisaient encore les possibilités sonores du système.

Aussi importantes que soient ces caractéristiques, elles ne représentent que la partie émergée de l'iceberg. L'existence d'un clavier alphabétique conduit à d'autres avantages, certains immédiats et d'autres plus profonds et plus riches liés à l'architecture du calculateur.

La première caractéristique est l'affichage "en clair", et non plus en code, des *fonctions programmées.* L'appellation de "langage machine spécialisé", utilisée jusqu'ici pour les calculatrices programmables, est maintenant inadéquate. Tout juste peut-on parler pour le HP-41 de langage d'assemblage spécialisé. En fait, en développant ce calculateur, HEWLETT-PACKARD a élaboré un langage évolué original, proche du FORTH, qui lui confère un ensemble de propriétés qui restent, à l'heure actuelle, uniques.

De la compréhension de ces possibilités dépend la façon dont on utilise le HP-41.

CE LANGAGE EST UN LANGAGE INTERPRETE.

Les instructions introduites dans la mémoire de la machine ne sont pas immédiatement compréhensibles par le microprocesseur. Avant d'être exécutées elles doivent être traduites et décomposées en une série (parfois très complexe) de micro-instructions qui peuvent, elles, être appréhendées par la "puce" du HP-41. Cette opération de décryptage se nomme l'interprétation.

LE HP-41 DISPOSE D'UN LANGAGE SYMBOLIQUE

Un ordinateur, quel qu'il soit, occupe les trois-quarts de son temps à rechercher des informations dans un espace de sa mémoire pour les transférer ailleurs. Il se peut qu'au passage ces informations soient modifiées mais ce n'est pas absolument nécessaire! Pour effectuer ces transferts le microprocesseur doit connaître les positions de départ et d'arrivée des informations, ce que l'on nomme les adresses absolues.

Il existe deux types d'informations :
- Les données : valeurs numériques ou groupe de caractères.
- Les instructions, dont l'enchaînement constitue un programme.

Au niveau du langage machine ces informations sont toutes numériques. Mais l'utilisateur moyen n'a rien d'un microprocesseur, il lui est plus facile de retenir des mots que des numéros ou même des séries d'instructions : les programmeurs préfèrent les symboles aux nombres. Le microprocesseur doit donc effectuer la tâche fastidieuse consistant à mettre en correspondance le symbole et l'adresse absolue pour atteindre l'information désignée. Il peut le faire par l'intermédiaire de catalogues comparables dans le principe à l'agenda qui vous permet quotidiennement de retrouver un n° de téléphone à partir du nom d'un individu. On peut donc mesurer le degré d'évolution d'un langage par son degré de symbolisme.

LE LANGAGE HP-41 EST MODULAIRE

Dans le monde de la micro-informatique le HP-41 est une des rares machines qui autorise l'implantation d'un nombre indéterminé de programmes. Chacun peut être créé, modifié, effacé et (si l'on dispose de périphérique de masse) lu ou écrit indépendamment.

A cette indépendance physique s'oppose une dépendance logique. Tout programme peut commander l'exécution d'un groupe d'instructions appartenant à un autre programme; il suffit que cette séquence débute par une étiquette alphabétique : LBL "X..." et s'achève par RTN ou END.

Il est alors possible de décomposer un programme complexe en une série de sous-unités (elles-mêmes décomposables en blocs plus simples) selon les sains principes de la programmation descendante. Les problèmes posés par la manipulation élémentaire des données sont ainsi renvoyés aux niveaux d'appel les plus profonds et ne viennent plus noyer l'architecture logique du programme clairement détaillée aux niveaux supérieurs. Cette façon de faire offre de multiples avantages:

- Il devient très difficile de commettre des erreurs de conception dans l'agencement des opérations nécessaires à la résolution d'un problème. Si malgré tout, cela advenait, cette erreur serait rapidement localisée parmi un nombre restreint d'instructions.

Il est également possible de tester individuellement chacun des blocs pour vérifier que les sorties obtenues correspondent bien aux valeurs espérées pour un jeu de données d'essais proposées en entrée.

Enfin certains de ces blocs peuvent être à l'expérience si utiles qu'on souhaite en réaliser une version en assembleur. Ce module en est l'illustration. La presque totalité des fonctions contenues dans le module "PANAME" ont été conçues comme des "procédures" indépendantes, initialement écrites en langage utilisateur et publiées en 1982. La conception du traitement des tableaux et des différentes fonctions disponibles maintenant datent pour la plupart de cette époque.

PROGRAMMATION SUR HP-41.

Il existe trois niveaux de programmation sur HP-41 :

- LES PROGRAMMES proprement dits, constitués en fait d'une succession de procédures éventuellement entrecoupées de tests. Toujours orienté vers la résolution d'un problème spécifique, le programme résume la part "stratégique" de l'art du programmeur. Compréhensible à la simple lecture, un programme doit avoir une documentation interne par le biais des étiquettes et des commentaires.

- LES PROCEDURES (ou sous-programmes de traitement). Elles représentent le versant "tactique" de la programmation. Une procédure est courte, rapide, elle économise au maximum l'espace programmable et altère le moins possible l'espace des variables. Elle correspond à l'exécution d'une tâche unique.

Elle est souvent suffisamment générale pour être utilisée de multiples fois par un même programme, voire par des progammes distincts. Elle est donc susceptible d'être maintenue en mémoire de façon permanente. Ce fait impose une standardisation des méthodes de programmation, source d'une importante économie d'effort. Si une procédure répond à ces critères, elle peut être considérée comme une nouvelle fonction du langage HP-41. Démonstration de l'ultime qualité de ce langage : sa plasticité, son évolutivité.

- LES FONCTIONS EN ASSEMBLEUR. Elles sont les éléments constitutifs du langage même. Une fonction doit être, encore plus qu'une procédure, d'usage général. Les deux auteurs de ce module ont réussi cette gageure de mettre à notre disposition un ensemble cohérent de plus de 120 fonctions.

UN PREMIER CONCEPT : LE TRAITEMENT DES PERIPHERIQUES

Il faut avoir expérimenté les fonctions de ce module pour apprécier la simplification qu'il apporte. Qu'il s'agisse des fonctions de la vidéo ou de celles des imprimantes, elles nous apportent un gain de temps à la programmation comme à l'exécution vraiment appréciable. En regard des séquences "escape" les instructions en "clair" nous font franchir une étape équivalente à lire "SIN" là où nous écrivions jadis "31 04". De même, écrire "CLEAR" au lieu de "27 ACCHR 69 ACCHR OUTA" est non seulement plus explicite, mais présente l'avantage considérable de fonctionner en mode trace. De ce point de vue le module "PANAME", apporte une facilité et résoud bien des difficultés jusque là insurmontables.

AUTRE CONCEPT : LE TRAITEMENT DE TABLEAUX

Combien de fois avons-nous grincé des dents en entendant affirmer que pour traiter des matrices le HP-41 n'était pas l'outil adapté. Aujourd'hui chacun pourra mesurer à quel point l'usage que nous préconisions de la pile opérationnelle nous préparait au développement de ces nouvelles fonctions. L'essor que va connaître le langage FORTH apportera une preuve supplémentaire, si tant est qu'elle soit nécessaire, à cette conception. Ceux qui auront eu la chance de s'exercer à la logique RPN se trouveront de plein-pied dans un langage qui relégue le BASIC au rang des antiquités.

Nous espérons que tous ceux qui ont pu développer avec plaisir des programmes sur HP-41 sauront reconnaître l'avantage qu'ils peuvent tirer de l'acquisition du module "PANAME".

PHILIPPE DESCAMPS

AVERTISSEMENT

Le module PANAME contient de nouvelles fonctions pour le HP-41, et les applications peuvent être nombreuses et diverses. Comme pour les autres fonctions du HP-41, aussi bien celles qui se trouvaient d'origine dans le calculateur que celles des modules qui ont été créés par la suite, seuls les utilisateurs par la multiplicité de leurs préoccupations sont à même de faire apparaître l'intérêt des fonctions mises à leur disposition.

Cependant il s'avère utile de réunir dans un recueil des exemples variés afin de favoriser le perfectionnement de chaque utilisateur. C'est une des raisons d'être du club PPC que de proposer un support à la communication des solutions de différents problèmes au plus grand nombre et participer ainsi à une formation permanente de chacun d'entre nous. De ce point de vue il est incontestable que nous devons beaucoup à notre club sans qui la création de ce module aurait été inimaginable.

Pour l'instant le module PANAME est donc fourni avec une notice explicative que nous avons essayée de rendre la plus claire et la plus précise possible. Pourtant nous reconnaissons nos limites et nous proposons de poursuivre ce travail avec chacun d'entre vous afin de constituer un document dont l'esprit serait proche de celui qui présida à la réalisation du module PPC.

Pour ceux d'entre vous qui désireraient dans quelque temps recevoir un livret d'applications pour le module PANAME, nous leur proposons de s'inscrire dès à présent auprès de J.J. DHENIN, BCMW 2 bis rue N. HOUEL 75005 PARIS. Dès que l'ouvrage sera disponible nous leur ferons parvenir.

Comment réaliserons-nous cet ouvrage collectif ?
Chacun d'entre vous va rencontrer dans le document actuel des points qui lui paraîtront nécessiter des éclaircissements. Dans la mesure du possible, nous attendons de vous des questions écrites parce qu'elles nous fourniront des interrogations dont nous garderons la trace afin de remettre à jour la notice du module PANAME. Dans la mesure du possible nous préférons que vous nous soumettiez des propositions de reformulation de tel ou tel passage. En effet, nous nous sommes tellement habitués à l'utilisation de nos nouvelles fonctions que nous ne sommes plus en mesure d'estimer les difficultés que vous pouvez rencontrer. C'est vous et vous seulement qui pouvez dire comment un manuel d'utilisation doit être rédigé.

Enfin, les exemples d'utilisation étant vraiment la meilleure explication d'une fonction, surtout lorsque le manuel est rédigé dans une langue à laquelle nous ne sommes pas habitués, nous attendons que vous nous fassiez part de vos propres applications, courtes si possible.

En fonction de vos suggestions et de nos propres travaux, nous serons en mesure de vous fournir prochainement un document plus complet encore et surtout plus près de vos préoccupations.

Heureuse programmation.

APERCU DES FONCTIONS DU MODULE

AID :
Retourne en X le numéro d'identification d'Accessoire du périphérique principal.

ID :
Retourne en ALPHA l'identification d'Appareil du périphérique principal.

FINDAID :
Recherche sur la boucle un périphérique d'AID (ou de classe) spécifié par X (<0 pour une classe) et renvoie son adresse en X.

OUTAX :
OUTA à répétition . |X| indique le nombre de répétitions.

OUTCR :
Envoie au périphérique principal un caractère 13 (retour chariot).

OUTLF :
Envoie au périphérique principal un caractère 10 (fin de ligne).

OUTLFX :
Envoie au périphérique principal un ou plusieurs caractères 10 (fin de ligne), |X| indique le nombre de caractères.

OUTSPX :
Envoie au périphérique principal un ou plusieurs caractères 32 (espace); |X| indique le nombre de caractères.

OUTXB :
Envoie au périphérique principal un octet spécifié par |X|.

OUTYBX :
Envoie une ou plusieurs fois au périphérique principal un octet spécifié par |Y|. Le nombre de répétitions est indiqué par |X|.

OUTa :
Similaire à OUTA, mais arme le bit 7 de tous les octets envoyés (par exemple pour la vidéo inverse sur HP82163).

OUTaX :
OUTa à répétition (cf OUTAX).

RCLSEL :
Retourne en X l'adresse du périphérique principal. Si le SELECT est > au nombre de périphériques, RCLSEL retourne 1.

Fonctions pour HP82163

CLEAR :
Efface l'écran.

CLEARO :
Efface l'écran à partir du curseur.

CSRDN :
Descend le curseur d'une position.

CSRHX :
Déplace le curseur de |X| positions horizontalement (à gauche si X<0, à droite dans le cas inverse)

CSRL :
Déplace le curseur d'une position vers la gauche.

CSROFF :
"Eteint" le curseur.

CSRON :
Fait apparaître le curseur.

CSRR :
Déplace le curseur d'une position vers la droite.

CSRVX :
Déplace le curseur de |X| positions verticalement (vers le haut si X<0, vers le bas dans le cas inverse).

CSRUP :
Déplace le curseur d'une position vers le haut.

CTYPE :
Sélectionne le type de curseur.

HOME :
Repositionne le curseur en (0,0).

SCRLDN :
Déroule l'affichage d'une ligne vers le bas.

SCRLUP :
Déroule l'affichage d'une ligne vers le haut.

SCRLX :
Déroule l'affichage selon |X|. X<0 vers le bas, X>0 vers le haut.

XYTAB :
Déplace le curseur à la position (|X|,|Y|).

Fonctions pour HP82162

CLBUF :
Efface la mémoire tampon.

8BIT :
Sélectionne le mode 8-BIT.

ESCAPE :
Sélectionne le mode ESCAPE.

PARSE :
Sélectionne le mode césure sur blanc.

STATUS :
Rapelle en X et Y les 2 octets d'état de l'imprimante.

TABCOL :
Effectue une tabulation absolue au niveau du point selon |X|.

UNPARSE :
Sélectionne le mode césure au 24ème caractère.

Fonctions pour HP82905

BELL :
Ding!

CHARSET :
Sélectionne le jeu de caractères : X=0 primaire; X=1 secondaire.

FFEED :
Effectue un saut de page.

FORMLEN :
Indique le nombre de lignes dans une page logique (selon |X|).

GRAPHX :
Indique à l'imprimante d'interpréter les |X| prochains caractères comme des codes de colonnes graphiques.

MODE :
Sélectionne le mode d'impression : 0 = Normal, 1 = Dilaté, 2 = Comprimé, 3 = Dilaté-Comprimé, 9 = Gras.

SKIPOFF :
Invalide la fonction de saut de perforations.

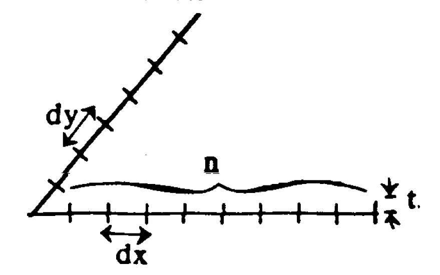
SKIPON :
Valide la fonction de saut de perforations.

TEXTLEN :
Indique le nombre de lignes de texte par page logique (selon |X|).

VSPAC :
Sélectionne l'espacement vertical entre 2 lignes; |X| indique le nombre de lignes par pouce.

Fonctions pour mini-plotter

AXIS :
Trace un axe au format suivant : t= dimension d'un demi tiret -/-
dy= distance entre 2 tirets -/- prise sur l'axe des y
dx= distance entre 2 tirets -/- prise sur l'axe des x
n= nombre de tirets



Les données étant fournies ainsi :
T = t, Z = dy, Y = dx, X = n.

BACKSP :
Recule d'un caractère.

BACKSPX :
Recule d'un caractère ou plus (|X| indique le nombre de caractères).

BOX :
Trace un rectangle dont 2 coins opposés ont pour coordonnées (x1,y1) et (x2,y2), avec T = y2, Z = x2, Y = y1, X = x1.

COLOR :
Selectionne la couleur du tracé.

***CSIZE :**
Sélectionne la taille des caractères.

***DRAW :**
Trace une ligne jusqu'au point de coordonnées (X,Y).

***HOME :**
Ramène le stylo en position (0,0).

***LABEL :**
Imprime le contenu du registre ALPHA (impression de textes en mode graphique; permet les directions définies par *LDIR).

***LDIR :**
Définit la direction d'écriture pour *LABEL.

***MOVE :**
Déplace le "stylo" jusqu'à la position (X,Y).

***LTYPE :**
Définit le type de ligne pour *DRAW et RDRAW (|X| = 0 à 15).

***PLREGX :**
Un pointeur bbb,eee étant placé en X, trace la ligne brisée passant par les points [(Rbbb),(Rbbb+1)], [(Rbbb+2),(Rbbb+3)] ....[(Reee-1), (Reee)].

RDRAW :
Trace une ligne jusqu'à la position (X,Y) relative à la position actuelle du "stylo".

RESET :
Initialisation; déplace le stylo jusqu'à la marge gauche et sélectionne le mode texte.

REVLF :
Fait revenir le papier en arrière d'une ligne.

REVLFX :
Fait revenir le papier en arrière d'une ligne ou plusieurs (selon |X|).

RMOVE :
Déplace le "stylo" jusqu'à la position (X,Y) relative à la position actuelle.

SETORG :
Redéfinit l'origine (0,0) comme étant la position actuelle du "stylo".

Utilitaires

/MOD :
Renvoie le quotient en Y et le reste en X de la division euclidienne. (c'est à dire calcule le quotient et le modulo de la division Y/X dans l'ensemble des entiers.)

AD-LC :
Renvoie les coordonnées (ligne, colonne) d'un élément d'un tableau connaissant son adresse.

ALENG :
Renvoie la longueur de la chaine présente en ALPHA.

ANUM :
Place dans X la première valeur numérique contenue dans la chaîne présente dans le registre ALPHA.

ANUMDEL :
Idem ANUM, et detruit le début de la chaîne jusqu'à cette valeur comprise.

APPX
Place à la suite de la chaîne presente dans le registre ALPHA la partie entière de la valeur présente en X. Le séparateur décimal n'est pas recopié.

AROT
Opere une rotation des caractères présents dans le registre ALPHA du nombre de positions indiquees dans X.

ATOXL
Place dans X le code décimal du caractère de gauche du registre ALPHA et supprime ce caractère du registre ALPHA.

ATOXR
Place dans X le code décimal du caractère de droite du registre ALPHA et supprime ce caractère du registre ALPHA.

ATOXX
Place dans X le code décimal du caractere dont la position est specifiee par le registre X.

BLDPT :
Construit un pointeur dans X à partir des éléments présents dans Z, Y et X. Si X>0, X = zzz,yyyxx . Si X<0, X = code de matrice tel que Z = premier registre du tableau, Y = nombre de lignes, |X| = nombre de colonnes.

BRKPT :
Décompose le registre X en trois valeurs numériques (inverse de BLDPT.)

CHFLAG :
Au cours de la construction du programme, l'utilisateur définit un état du HP-41 en utilisant les instructions habituelles (en mode calcul). En mode programme XEQ "CHFLAG" place dans le programme 2 lignes :

01 CHFLAG
02 "...." une chaîne alpha non habituelle.

Au cours de l'exécution du programme les 2 lignes configureront le calculateur conformément à la situation définie au moment de la programmation.

CLINC :
Efface dans le registre X les incréments (ie. à partir du 4è chiffre après la virgule) .

COLPT :
Construit le code de la colonne d'une matrice connaissant son numéro en Y et le code de la matrice en X.

GETRGX :
Copie dans les mémoires spécifiées par X, les registres du fichier de données en se conformant aux 2 incréments.

LC-AD :
Renvoie l'adresse (numéro de registre) d'un élément d'une matrice connaissant le numéro de ligne et le numéro de colonne.

LINPT :
Construit le code de la ligne d'une matrice connaissant son numéro et le code de la matrice.

NOP :
N'effectue aucune opération.

OUT :
Préfixe destiné à faciliter l'écriture des fonctions.

POSA :
Renvoie la position dans la chaîne ALPHA du caractère spécifié dans X.

PSIZE :
Alloue aux données le nombre de registres indiqués dans X.

READEM :
Copie de la cassette dans les XMEMORY le "fichier de mémoire d'extension 'désigné' par le registre ALPHA. cf WRTEM.

RG :
Préfixe pour aider à la frappe des fonctions.

RG+- :
Effectue l'addition (ou la soustraction) terme à terme des registres désignés par les codes placés en Y et en X.

RG* :
cf RG+-, effectue une multiplication.

RG/ :
cf RG+-, effectue une division.

RG+Y, RG*Y et RG/Y :
effectuent une opération arithmétique des éléments désignés par X, avec l'opérande placé en Y.

RGAX :
Si X<0, recopie le registre ALPHA dans les registres de données par bloc de 6 caractères, Si X>0, place à droite de la chaîne placée en ALPHA le contenu des registres de données à partir de la mémoire désignée en X, et jusqu'à la fin de la chaîne telle qu'elle avait été chargée par RGAX.

RGCOPY :
Si X>0 copie les registres désignés par X dans les registres spécifiés en Y.
Si X<0, les blocs sont échangés.
Admet un incrément.

RGINIT :
Si X>0, Place la valeur 0 dans les registres désignés par X. Si X<0, place les nombres de 1 à n dans les registres.

RGNb :
Renvoie le nombre d'éléments déterminés par un code ddd,fffii en X.

RGORD :
Remplace chaque valeur, contenue dans les registres de données spécifiés, par leur rang.

RGSUM :
Renvoie la somme des valeurs spécifiées par le code en X. Si X<0, calcule la somme des valeurs absolues.

RGVIEW :
Entrée ou affichage des registres. Détails nécessaires.

SAVERGX :
Fonction inverse de GETRGX. Recopie les registres désignés en X, dans le fichier de données courant, à partir du pointeur et en suivant l'incrément j :
X = bbb,eeeiijj

SIZE? :
Renvoie le nombre de registres alloués aux données.

SORT :
Trie par ordre croissant (X>0) ou décroissant (X<0) les valeurs des registres désignés en X. Tri alpha et numérique.

STO>L :
Copie la valeur placée en X à l'adresse spécifiée en L, et incrémente L

SUB$:
Extraction et/ou justification d'une sous-chaîne.

TF55 :
Valide ou invalide l'imprimante.

VKEYS :
Catalogue des assignations.

WRTEM :
Crée et remplit un fichier de mémoire d'extension sur la cassette. Ce fichier est un "WRTA" de la mémoire d'extension.

Y/N :
Simplifie les programmes qui, lors de leur utilisation, posent à l'utilisateur une question à laquelle il faut répondre par OUI ou par NON.

X...NN

AID (Accessory **ID**entity) permet de déterminer l'Identification d'Accessoire de l'appareil principal. L'Identification d'Accessoire est un nombre entre 0 et 255 qui identifie le type de l'appareil.

Par exemple, l'Identification d'Accessoire de l'imprimante thermique HP82162A est 32. Si l'appareil principal est une imprimante HP82162A, la fonction AID place dans le registre X le nombre 32.

INSTRUCTIONS COMPLETES POUR AID

La fonction AID rappelle dans le registre X, après montée de la pile si elle est autorisée, un nombre représentant l'Identification d'Accessoire de l'appareil principal. Pour connaitre la valeur correspondant à un appareil donné, se referer à la description du message HPIL "Envoi Identification Accessoire" ("Send Accessory Identity") dans le manuel de l'appareil.

Si l'appareil principal ne possède pas d'Identification d'Accessoire, le message d'erreur NO RESPONSE est affiché.

Fonctions associées :
dans le module **I/O** : FINDAID, ID
dans le module **HPIL** : FINDID, SELECT, AUTOIO, MANIO
dans le module **PANAME** : RCLSEL

Appendice C

Ce tableau indique, pour chacune des classes d'appareils : son nom, l'intervalle de valeurs d'Identification d'Accessoire correspondant ainsi que le nombre ("identificateur de classe") à placer dans le registre X pour la recherche d'un appareil de la classe concernée par la fonction FINDAID.

Classe	AID	Identificateur de classe
Contrôleur	0 à 15	-1
Stockage de masse	16 à 31	-16
Imprimante	32 à 47	-32
Affichage	48 à 63	-48
Interface	64 à 79	-64
80 à 95		-80
Appareil graphique	96 à 111	-96
112 à 127		-112
128 à 143		-128
144 à 159		-144
160 à 175		-160
176 à 191		-176
192 à 207		-192
208 à 223		-208
224 à 239		-224
240 à 255		-240

ID (Device **ID**entity) permet de déterminer l'Identification d'Appareil de l'appareil principal. L'Identification d'Appareil est une chaine de caractères qui identifie l'appareil, et qui indique généralement le fabricant et la référence de l'appareil.

Par exemple, l'Identification d'Appareil de l'interface HPIL-RS232C est "HP82164A". Si l'appareil principal est une interface HPIL-RS232C, la fonction ID place dans le registre ALPHA la chaîne de caractères HP82164A.

INSTRUCTIONS COMPLETES POUR ID

La fonction ID place dans le registre ALPHA l'Identification d'Appareil de l'appareil principal. Pour connaître la chaîne correspondant à un appareil donné, se référer à la description du message HPIL "Envoi Identification Appareil" ("Send Device Identity") dans le manuel de l'appareil.

Si l'appareil principal ne possède pas d'Identification d'Appareil, le message d'erreur NO RESPONSE est affiché.

Fonctions associées :
dans le module **HPIL** : FINDID, SELECT, AUTOIO, MANIO
dans le module **PANAME** : AID, FINDAID, RCLSEL

[OUT] est une fonction destinée à faciliter l'entrée au clavier de noms de fonctions commençant par OUT. Cette fonction s'utilise essentiellement assignée à une touche. Par exemple, assignée OUT à la touche [LN].

ASN "OUT" 15

(appuyez sur [ ] [ASN] [ALPHA] [O] [U] [T] [ALPHA] [LN]). Placez le calculateur en mode USER. Dès lors, pour exécuter ou programmer une fonction dont le nom commence par OUT, (par exemple OUTAX), appuyer sur :

[OUT] (touche LN) [ALPHA] A X [ALPHA]

Cette séquence est équivalente à :

[XEQ] [ALPHA] [O] [U] [T] [A] [X] [ALPHA]

Vous économisez par conséquent 3 pressions de touches à chaque utilisation d'une fonction dont le nom commence par les trois lettres OUT.

INSTRUCTIONS COMPLETES POUR OUT

1) Assignez [OUT] à une touche et placez le calculateur en mode USER.

2) Pour exécuter ou programer une fonction dont le nom commence par OUT, appuyez successivement sur :

[OUT] (touche assignée précédemment)

[ALPHA]

.....caractères du nom de la fonction,
.....non compris les 3 premiers
.....(par ex. YBX pour la fonction OUTYBX).

[ALPHA]

OUTAX effectue une ou plusieurs fois l'opération OUTA, envoi du contenu du registre ALPHA à l'appareil principal. La valeur absolue du contenu du registre X indique le nombre de OUTA désiré.

Si le drapeau 17 est désarmé, un indicateur de fin de ligne (caractères CR et LF, de codes décimaux respectifs 13 et 10) est envoyé à la suite de la chaîne ALPHA sur la boucle HPIL à chaque envoi.

Si le drapeau 17 est armé, la chaîne est simplement envoyée plusieurs fois, sans caractères de séparation.

INSTRUCTIONS COMPLETES POUR OUTAX

Placer la chaîne à envoyer plusieurs fois dans le registre ALPHA, le nombre d'"envois" de la chaîne dans le registre X, et positionner le drapeau 17 suivant l'option désirée (cf. plus haut) et exécuter OUTAX.

EXEMPLE D'UTILISATION DE OUTAX

Pour tracer une ligne de séparation constituée de 40 chaînes "-*" sur une imprimante HP82905B, utiliser la séquence (on suppose que l'imprimante a été déclarée appareil principal par des opérations antérieures) :

"-*" SF 17 40 OUTAX ADV

Fonctions associées :
Toutes les fonctions commençant par OUT et dans l'HPIL, MANIO et SELECT qui sont nécessaires à la selection de l'appareil.

OUTCR

- Envoi du caractère CR ("retour chariot") -

OUTCR (**OUT**put Carriage **R**eturn) envoie à l'appareil principal un caractère CR ("retour chariot", code décimal 13).

OUTLF

- Envoi du caractère LF ("saut de ligne") -

OUTLF (**OUT**put Line Feed) envoie à l'appareil principal un caractère LF ("saut de ligne", code décimal 10)

OUTLFX

- Envoi d'un ou plusieurs caractères LF ("saut de ligne") -

OUTLFX (**OUT**put Line Feeds by X) envoie à l'appareil principal un ou plusieurs caractères LF ("saut de ligne", de code décimal 10), le nombre de caractères étant spécifié par la valeur absolue du contenu du registre X (qui doit appartenir à l'intervalle 0 à 999).

INSTRUCTIONS COMPLETES POUR OUTLFX

Placer le nombre de caractères LF à envoyer dans le registre X et exécuter OUTLFX.

OUTSPX (OUTput SPaces by X) envoie à l'appareil principal un ou plusieurs caractères "espace" (de code décimal 32), le nombre de caractères étant spécifié par la valeur absolue du contenu du registre X (qui doit appartenir à l'intervalle 0 à 999).

INSTRUCTIONS COMPLETES POUR OUTSPX

Placer le nombre de caractères "espace" à envoyer dans le registre X et exécuter OUTSPX.

EXEMPLE D'UTILISATION DE OUTSPX

Exemple: De nombreuses imprimantes ne disposent pas d'instruction de tabulation. La fonction OUTSPX permet de simuler très facilement une telle instruction. A titre d'illustration, le court programme suivant OUTAT envoie à l'appareil principal une chaîne de longueur fixée L constituée du contenu du registre ALPHA complété par des espaces en nombre suffisant. Si la longueur de la chaîne présente dans le registre ALPHA est supérieure à L, ladite chaîne est tronquée à L caractères (1).

La longueur maximale du contenu du registre ALPHA limite l'utilisation de ce programme à des chaînes d'au plus 24 caractères.

Mode d'emploi du programme **OUTAT** :

- Placer le nombre L dans le registre X ;
- Placer la chaîne à envoyer dans le registre ALPHA ;
- Exécuter **OUTAT**.

Ce programme détruit les contenus des registres X, T, et LASTx. Il arme en outre le drapeau 17.

Remarque importante : Le paramètre L placé en entrée dans le registre X doit être un entier positif supérieur ou égal à 1.

Listing du programme **OUTAT** :

```
LBL "OUTAT"  ALENG  X>Y?  GTO 01
-  LBL 02  SF 17  OUTA  OUTSPX  RTN
LBL 01  DSE Y  NOP  CLX  1 E2  /  SUB$
CLX  GTO 02  END
```

N.B. Le texte est cadré à gauche. Pour obtenir un texte cadré à droite, il suffit de permuter les instructions OUTA et OUTASPX .

- Envoi d'un caractère donné par son code décimal - OUTXB

OUTXB envoie à l'appareil principal un caractère dont le code décimal est spécifié par la valeur absolue du contenu du registre X. Cette valeur doit appartenir à l'intervalle 0-255.

INSTRUCTIONS COMPLETES POUR OUTXB

Placer le code du caractère à envoyer dans le registre X et exécuter OUTXB.

EXEMPLES D'UTILISATION DE OUTXB

Exemple : Pour envoyer à l'appareil principal le caractère "\" (barre oblique inversée, de code décimal 92), utiliser la séquence: 92 OUTXB.

- Envoi d'un ou plusieurs caractères donnés par leurs codes - OUTYBX

OUTYBX envoie à l'appareil principal un ou plusieurs caractères identiques dont le code décimal est spécifié par la valeur absolue du contenu du registre Y. La valeur absolue du contenu du registre X spécifie le nombre de caractères à envoyer.

Restrictions : 0<=ABS(X)<=999 et 0<=ABS(Y)<=255

INSTRUCTIONS COMPLETES POUR OUTYBX

Placer le code du caractère à envoyer dans le registre Y, le nombre d'exemplaires à envoyer dans le registre X et exécuter OUTYBX.

EXEMPLES D'UTILISATION DE OUTYBX

Exemple 1 : Pour envoyer à l'appareil principal une série de vingt caractères "'" (apostrophe, de code décimal 39), utiliser la séquence:

39 ENTER^ 20 OUTYBX.

Exemple 2 : 'FMTNBZ' (ForMaT de NomBre avec Zéro)
Ce programme simule un format d'impression des nombres sans suppression des zéros de tête. Il s'utilise comme suit:

- Placer le nombre à imprimer dans le registre X ;
- Placer dans le registre Y le nombre de caractères maximum pouvant être occupés par le nombre (taille de la zone d'impression recevant le nombre) ;
- Sélectionner le mode d'affichage désiré ;
- Exécuter FMTNBZ.

Si le nombre de caractères nécessaires pour représenter le contenu du registre X est supérieur à la valeur placée dans le registre Y, la zone correspondante est remplie de caractères "*".

Après exécution, les contenus des registres X, Y, LASTx et ALPHA sont perdus.

Listing du programme FMTNBZ :

```
LBL "FMTNBZ"  CLA  ARCL X  X<0?  XEQ 00
CLX  ALENG  X>Y?  GTO 01  -  48  X<>Y  OUTYBX  OUTA  RTN
LBL 00  CLX  ATOXL  OUTXB  RTN
LBL 01  CLX  42  X<>Y  OUTYBX  END
```

OUTa fonctionne comme OUTA, avec la différence suivante : le bit n°7 de chaque octet envoyé est forcé à 1 (en d'autres termes, la valeur 128 est ajoutée aux codes des caractères qui ne sont pas déjà supérieurs ou égaux à 128), à l'exception toutefois des deux octets qui constituent l'indicateur de fin de ligne (caractères CR et LF, de codes décimaux respectifs 13 et 10) qui est envoyé à la fin de la chaîne ALPHA lorsque le drapeau 17 est désarmé.

INSTRUCTIONS COMPLETES POUR OUTa

Placer la chaîne à envoyer dans le registre ALPHA, positionner le drapeau 17 suivant l'option désirée (cf. plus haut) et exécuter OUTa.

EXEMPLES D'UTILISATION DE OUTa

Exemple 1 : Pour afficher une chaîne de caractères en mode "vidéo inversée" sur l'interface vidéo HP82163, il faut ajouter 128 aux codes des caractères de la chaîne avant de l'envoyer à l'interface. Cette opération est réalisée automatiquement par la fonction OUTa. Pour afficher une chaîne de caractères en "vidéo inversée", il vous suffit de déclarer l'interface vidéo comme appareil principal, de placer la chaîne désirée dans le registre ALPHA et d'exécuter OUTa. L'état du drapeau 17 détermine l'envoi ou le non-envoi d'un indicateur de fin de ligne.

Exemple 2 : Certaines imprimantes disposent d'une possibilité de soulignement automatique des caractères, qui nécessite l'ajout de 128 au code de chaque caractère dont on désire qu'il soit souligné. La fonction OUTa permet de simplifier énormément l'opération de soulignement avec de telles imprimantes.

Exemple 3 : L'accès aux caractères spéciaux dont dispose l'imprimante HP82905B peut se faire de deux manières :

- En utilisant le mode "jeu de caractères secondaires" qui donne aux codes de caractères compris entre 32 et 127 de nouvelles significations ;
- En utilisant des codes de caractères supérieurs 0 127.

Cette deuxième possibilité est d'un emploi particulièrement simple grâce à la fonction OUTa.

RCLSEL place dans le registre X, après montée de la pile si elle est autorisée, un nombre entier représentant l'adresse HPIL de l'appareil principal. En outre, RCLSEL vérifie l'intégrité de la boucle, c'est-à-dire qu'il n'y a pas de rupture électrique de la boucle et que tous les appareils sont sous tension (l'effet sur les appareils possédant un mode "STANDBY" est similaire à celui de la fonction PWRUP, voir la description de cette fonction dans le manuel du module HPIL HP82160A). Contrairement à la fonction de même nom du module "Extended I/O" HP82183A, la fonction RCLSEL du module PANAME retourne une valeur qui peut être différente de la dernière adresse spécifiée par la fonction SELECT. Cela se produit lorsque l'utilisateur spécifie un paramètre pour SELECT qui est supérieur au nombre d'appareils présents sur la boucle ; dans ce cas, l'adresse retournée par RCLSEL est égale à 1. Cette caractéristique est particulièrement appréciée dans des programmes comportant une boucle exécutée une fois pour chaque appareil de la boucle. Un simple test comparant l'adresse de sélection de l'appareil à traiter et la valeur retournée par RCLSEL permet de savoir si tous les appareils ont été traités. A titre d'illustration, les programmes LOOP (donné comme exemple d'utilisation de AID et ID) et FNDAIDN (donné comme exemple d'utilisation de FINDAID) utilisent cette méthode.

INSTRUCTIONS COMPLETES POUR RCLSEL

Exécuter RCLSEL ; un nombre représentant l'adresse de l'appareil principal est rappelé dans le registre X conformément aux conventions indiquées plus haut.

EXEMPLE D'UTILISATION DE RCLSEL

RCLSEL peut être utilisé dans un programme qui modifie la sélection de l'appareil principal pour permettre la restauration de cette sélection en fin de programme. La valeur initiale de l'adresse de l'appareil principal est stockée dans un registre en début de programme par la séquence RCLSEL STO nn et restaurée au moment voulu par RCL nn SELECT.

Groupe 82163

Les fonctions de ce groupe sont destinées à faciliter l'utilisation de l'Interface Vidéo HP82163. Elles vous donnent la possibilité de contrôler complètement l'interface sans vous soucier des séquences d'échappement ou des caractères de contrôle requis par l'interface pour effectuer une opération, telle que l'effacement de l'affichage ou le déplacement du curseur vers le bas. A titre d'exemple, vous disposez pour ces deux opérations des fonctions CLEAR et CSRDN (Cursor Down), respectivement.

Toutes ces fonctions nécessitent que l'interface soit déclarée appareil principal. Référez-vous au mode d'emploi des fonctions FINDAID (dans ce manuel) et FINDID (dans le manuel du module HPIL HP82160A) pour connaître les différentes méthodes de sélection d'un appareil particulier.

En mode AUTOIO, si l'appareil principal ne possède pas une Identification d'Accessoire égale à 48 (Interface Vidéo standard), le message d'erreur AID ERR est affiché.

Néanmoins, les fonctions n'effectuent pas cette vérification en mode MANIO, ce qui permet de les utiliser par exemple avec l'interface vidéo Mountain Computer MC00701A (dont l'Identification d'Accessoire vaut 50), l'interface PAC-TEXT (48) ou l'interface MINITEL de KRISTAL(*). (48)

Pour plus de détails techniques concernant les séquences envoyées par ces fonctions, se référer à l'Appendice V.

* KRISTAL chemin des Clos Zirst 38240 MEYLAN (FRANCE), systèmes informatiques interface et applications techniques, instrumentation, est OEM agréé HP.

- Effacement de l'affichage -

CLEAR

CLEAR efface l'affichage, place le curseur à la position (0,0) et sélectionne le curseur de remplacement.

INSTRUCTIONS COMPLETES POUR CLEAR

Exécuter CLEAR.

EXEMPLE D'UTILISATION DE CLEAR

La séquence ESC E envoyée par la fonction CLEAR est utilisée par l'imprimante HP82905B comme séquence de "réinitialisation". La fonction CLEAR peut donc être utilisée pour réinitialiser cette imprimante, le mode MANIO étant toutefois nécessaire pour empêcher la fonction CLEAR de vérifier que l'Identification d'Accessoire de l'appareil principal est égale à 48.

- Effacement de l'affichage à partir du curseur -

CLEARO

CLEARO efface l'affichage à partir de la position du curseur jusqu'en bas de l'écran. La position du curseur et le type de curseur ne sont pas modifiés.

INSTRUCTIONS COMPLETES POUR CLEARO

Exécuter CLEARO.

CSRDN

- Déplacement du curseur vers le bas -

CSRDN (CurSoR DowN) déplace le curseur d'une position vers le bas. Si le curseur se trouve sur la dernière ligne de l'affichage, le curseur n'est pas déplacé.

CSRHX

- Déplacement horizontal du curseur -

CSRHX (Move CurSoR Horizontally by X) déplace le curseur horizontalement, d'un nombre de positions spécifié par la valeur absolue du contenu du registre X, et dans la direction spécifiée par le signe du contenu du registre X : X<0 pour un déplacement vers la gauche, X>=0 pour un déplacement vers la droite. Par exemple, -1 CSRHX est équivalent à CSRL et 1 CSRHX est équivalent à CSRR.

INSTRUCTIONS COMPLETES POUR CSRHX

Placer dans le registre X la valeur indiquant le déplacement désiré conformément aux indications ci-dessus et exécuter CSRHX.

CRSL

- Déplacement du curseur vers la gauche -

CSRL (CurSoR Left) déplace le curseur d'une position vers la gauche. Si le curseur se trouve sur la première colonne d'une ligne, le curseur est déplacé jusqu'à la dernière colonne de la ligne précédente, sauf s'il se trouve à la position (0,0), auquel cas il n'est pas déplacé.

CSROFF

- Suppression du curseur -

CSROFF supprime l'affichage du curseur. Le curseur n'est plus visible jusqu'à la prochaine exécution de CLEAR ou CSRON ou la prochaine initialisation de l'interface (mise sous tension ou messsages HPIL Appareil Libre -DCL- ou Appareil Sélecté Libre -SDC-).

CSRON

- Affichage du curseur -

CSRON active l'affichage du curseur. Cet affichage peut être supprimé par exécution de la fonction CSROFF.

CSRR

- Déplacement du curseur vers la droite -

CSRR (CurSoR Right) déplace le curseur d'une position vers la droite. Si le curseur se trouve sur la dernière colonne d'une ligne, le curseur est déplacé jusqu'à la première colonne de la ligne suivante, sauf s'il se trouve à la position (31,15), auquel cas il n'est pas déplacé.

- Déplacement vertical du curseur - CRSVX

CSRVX (Move **CurSoR** Vertically by **X**) déplace le curseur verticalement, d'un nombre de positions spécifié par la valeur absolue du contenu du registre X, et dans la direction spécifiée par le signe du contenu du registre X : X<0 pour un déplacement vers le haut, X>=0 pour un déplacement vers le bas. Par exemple, -1 CSRVX est équivalent à CSRUP et 1 CSRVX est équivalent à CSRDN.

INSTRUCTIONS COMPLETES POUR CSRVX

Placer dans le registre X la valeur indiquant le déplacement désiré conformément aux indications ci-dessus et exécuter CSRVX.

- Déplacement du curseur vers le haut - CSRUP

CSRUP (**CurSoR UP**) déplace le curseur d'une position vers le haut. Si le curseur se trouve sur la première ligne de l'affichage, le curseur n'est pas déplacé.

- Sélection du type de curseur - CTYPE

CTYPE (Cursor **TYPE**) sélectionne le type de curseur suivant la valeur présente dans le registre X :

- Pour X=0, sélectionne le curseur "d'insertion" (flèche clignotante) ;
- Pour X=1 ou -1, sélectionne le curseur de "remplacement" (pavé clignotant).

INSTRUCTIONS COMPLETES POUR CTYPE

Placer la valeur correspondant au type de curseur désiré dans le registre X et exécuter CTYPE. Noter qu'en cas d'utilisation avec l'Interface Vidéo Mountain Computer MC00701A, la sélection du curseur d'insertion (caractère de soulignement clignotant) ne valide pas le mode "insertion de caractères" ou "insertion de ligne".

- Retour du curseur en haut et à gauche de l'affichage - HOME

HOME déplace le curseur jusqu'à la position (0,0).

- Déroulement de l'affichage vers le bas - SCRLDN

SCRLDN (**SCRolL DowN**) déroule l'affichage d'une ligne vers le bas (c'est à dire fait disparaître la ligne inférieure de l'affichage et en fait apparaître une autre en haut de l'écran).

- Déroulement de l'affichage vers le haut - SCRLUP

SCRLUP (**SCRolL UP**) déroule l'affichage d'une ligne vers le haut (c'est-à-dire fait disparaître la ligne supérieure de l'affichage et en fait apparaître une autre en bas de l'écran).

- Déroulement de l'affichage paramétré par X - SCRLX

SCRLX (**SCRolL** as specified by **X**) déroule l'affichage d'un certain nombre de lignes vers le haut ou vers le bas, suivant la valeur contenue dans le registre X.

- Pour X<0, SCRLX déroule l'affichage de (-X) lignes vers le haut (correspond à l'exécution répétée de SCRLUP) ;
- Pour X>=0, SCRLX déroule l'affichage de X lignes vers le bas (correspond à l'exécution répétée de SCRLDN).

INSTRUCTIONS COMPLETES POUR SCRLX

Placer dans le registre X la valeur correspondant au déroulement désiré, et exécuter SCRLX.

- Déplacement du curseur à une position (X,Y) - XYTAB

XYTAB (**(X,Y) TAB**ulate) déplace le curseur à la position (x,y), le numéro de colonne x étant spécifié par la valeur absolue du contenu du registre X et le numéro de ligne y étant spécifié par la valeur absolue du contenu du registre Y.

INSTRUCTIONS COMPLETES POUR XYTAB

Placer dans le registre X le numéro de colonne, dans le registre Y le numéro de ligne et exécuter XYTAB.

Appendice V

Séquences envoyées à l'appareil principal par les fonctions du groupe 82163 FCNS.

"ESC" représente le caractère "échappement", de code décimal 27.

Fonction(s)	Séquence	Codes de caractères
CLEAR	ESC E	27 69
CLEARO	ESC J	27 74
CSRDN, CSRVX pour X>=0	ESC B	27 66
CSRL, CSRHX pour X<0	BS	08
CSROFF	ESC <	27 60
CSRON	ESC >	27 62
CSRR, CSRHX pour X>=0	ESC C	27 67
CSRUP, CSRVX pour X<0	ESC A	27 65
CTYPE pour X=0	ESC Q	27 81
CTYPE pour X=1 ou -1	ESC R	27 82
HOME	ESC H	27 72
SCRLDN, SCRLX pour X>=0	ESC T	27 84
SCRLUP, SCRLX pour X<0	ESC S	27 83
XYTAB	ESC % {c} {l}	27 37 col lgn

Groupe 82162

Les fonctions de ce groupe sont destinées à faciliter l'utilisation de l'Imprimante Thermique HP82162A. Ces fonctions vous permettront de tirer parti de toutes les possibilités de cette imprimante, dont la plupart ne sont pas explicitées dans le manuel.

Ces caractéristiques sont :

- deux jeux de caractères différents ;
- un mode "césure sur blanc" ;
- une possibilité de tabulation au niveau du point, indépendamment des données, éventuellement déjà présentes dans la mémoire-tampon d'impression ;
- une possibilité de connaître de façon précise l'état de l'imprimante.

Ces fonctions opèrent sur la première imprimante HP82162A située sur la boucle à partir de l'appareil principal. Si aucune imprimante HP82162A n'est trouvée sur la boucle, le message d'erreur NO 82162 est affiché. La seule exception à cette règle concerne la fonction STATUS du module PANAME ; consultez le mode d'emploi de cette fonction pour plus d'information.

8BIT

- Sélection du mode "huit bits" -

8BIT sélectionne le mode "huit bits", ce qui valide le jeu de caractères "HP41". Ce mode est automatiquement sélectionné lors de l'exécution de toute fonction d'impression spécifique (fonction figurant sous l'en-tête -PRINTER 2E du module HPIL). Cette fonction n'est donc utile que si l'on utilise l'imprimante HP82162A avec des fonctions non spécifiques à l'impression, telles que OUTA ou OUTYBX.

ESCAPE

- Sélection du mode "échappement" -

ESCAPE sélectionne le mode "échappement", ce qui rend actif le jeu de caractères ASCII (non-HP41). Dans ce mode, l'envoi de caractères à l'imprimante ne peut se faire par les fonctions spécifiques à l'impression car celles-ci resélectionnent le mode "huit bits". Néanmoins certaines applications peuvent nécessiter l'utilisation du jeu de caractères ASCII. La fonction ESCAPE permet l'emploi de ce jeu, l'impression devant alors être obligatoirement réalisée par la fonction OUTA ou des fonctions associées qui n'ont pas d'autre action que l'envoi de caractères à l'appareil principal. Noter que dans ce cas, l'imprimante doit être déclarée appareil principal, ce qui n'est pas indispensable avec les fonctions spécifiques à l'impression telles que PRA.

PARSE

- Sélection du mode "césure sur blanc" -

PARSE sélectionne le mode "césure sur blanc", qui permet l'impression automatique de textes sans coupure des mots en fin de ligne. Un retour à la ligne est généré par l'imprimante au niveau de l'espace séparant deux mots si le mot qui suit l'espace ne peut être imprimé entièrement sur la ligne en cours.

CLBUF

- Effacement mémoire tampon -

CLBUF place l'imprimante dans les mêmes conditions qu'à la mise sous tension, c'est-à-dire :

- le chariot est à droite ;
- la mémoire tampon d'impression est vide ;
- les modes actifs sont : "échappement", largeur simple, majuscule, justification à gauche et césure au 24ème caractère.

Cette fonction est principalement utilisée pour vider la mémoire tampon des données pouvant éventuellement s'y trouver, cette opération étant impossible autrement.

- Sélection du mode "césure au vingt-quatrième caractère"

UNPARSE

UNPARSE supprime le mode d'impression spécial dit "césure sur blanc" sélectionné par la fonction PARSE.

- Tabulation par colonnes -

TABCOL

[TABCOL] permet d'effectuer une tabulation absolue au niveau de la colonne de points (par oppositiona SKPCOL qui effectue une tabulation relative).

En utilisant TABCOL, on peut facilement imprimer des tableaux à plusieurs colonnes (FMT ne permet que deux colonnes).

INSTRUCTIONS COMPLETE POUR TABCOL

Placer le numéro de la colonne de la tabulation désirée (de 0 à 167) dans le registre X et exécuter TABCOL.

EXEMPLE D'UTILISATION DE TABCOL

Pour imprimer le tableau suivant :

```
A=    123.00  FF
B=     23.95  FS
C=   1115.70  FB
```

on peut utiliser la séquence :

```
FIX 2  CLBUF  "A="  ACA  28  TABCOL  123  ACX  91  TABCOL  "FF"  ACA  PRBUF
              "B="  ACA  28  TABCOL  23.95  ACX  91  TABCOL  "FS"  ACA  PRBUF
              "C="  ACA  28  TABCOL  1115.7  ACX  91  TABCOL  "FB"  ACA  PRBUF
```

Groupe 82905 FCNS

Les fonctions de ce groupe sont destinées à faciliter l'utilisation de l'Imprimante 80 colonnes HP82905B. Elles vous donnent la possibilité de contrôler complètement l'imprimante sans connaître explicitement les séquences d'échappement et les caractères de contrôle requis par l'imprimante pour l'exécution d'une tâche déterminée. Ces fonctions augmentent considérablement la facilité d'écriture et la lisibilité de programmes mettant en oeuvre les nombreux modes de fonctionnement de l'imprimante HP82905B.

Toutes ces fonctions nécessitent que l'imprimante soit déclarée appareil principal. Référez-vous au mode d'emploi des fonctions FINDAID (dans ce manuel) et FINDID (dans le manuel du module HPIL HP82160A) pour connaître les différentes méthodes de sélection d'un appareil particulier.
En mode AUTOIO, si l'appareil principal ne possède pas une Identification d'Accessoire égale à 33, le message d'erreur AID ERR est affiché.
Néanmoins, les fonctions n'effectuent pas cette vérification en mode MANIO, ce qui permet leur emploi avec d'autres imprimantes utilisant les mêmes séquences d'échappement et caractères de contrôle que l'imprimante HP82905B.

Pour plus de détails techniques concernant les séquences envoyées par ces fonctions, se référer à l'Appendice P.

BELL

- Signal sonore -

BELL actionne le signal sonore de l'imprimante pendant une seconde. Cette fonction peut être utilisée par exemple pour prévenir l'utilisateur d'une condition particulière nécessitant son intervention.

CHARSET

- Sélection du jeu de caractères -

CHARSET sélectionne le jeu de caractères primaire si X=0, le jeu de caractères secondaire si X=1. Se référer au Manuel d'utilisation de l'imprimante HP82905B pour la liste complète des deux jeux de caractères.

FFEED

- Saut de page -

FFEED envoie à l'imprimante une commande 'saut de page' qui fait avancer le papier jusqu'en haut de la page suivante. Notez que vous devez positionner le papier correctement et définir le nombre de lignes par page correspondant au papier utilisé (au moyen de la fonction FORMLEN) pour que FFEED avance le papier effectivement jusqu'à la page suivante.

FORMLEN

- Longueur de page -

FORMLEN définit le nombre de lignes dans une page (ce nombre dépend du type de papier utilisé et de l'espacement entre les lignes choisi par la fonction VSPAC).

Le contenu du registre X (sans tenir compte du signe éventuel) indique le nombre de lignes désiré, qui doit être compris entre 1 et 128. A la mise sous tension ou lors d'une réinitialisation par la fonction CLEAR (voir le mode d'emploi de cette fonction pour la description de cette possibilité), le nombre de lignes utilisé par l'imprimante est 66.

GRAPHX

- Génération de graphiques -

GRAPHX indique à l'imprimante d'interpréter les octets suivants non comme des caractères, mais comme des données binaires, chaque valeur correspondant à une colonne de points. Consultez le Manuel d'utilisation de l'imprimante pour la correspondance entre les valeurs reçues et l'impression résultante (paragraphe "Mode Graphique").

Le contenu du registre X indique combien d'octets l'imprimante doit interpréter comme des données graphiques (le signe de X est ignoré).

- Mode d'impression - MODE

MODE définit le mode d'impression selon la valeur présente dans le registre X (le signe étant ignoré) conformément au tableau suivant :

Contenu de X :	Mode :	Nb car. / ligne :
0	Normal	80
1	Dilaté	40
2	Comprimé	132
3	Dilaté-gras	66
9	Gras	80

Vous pouvez mélanger sur la même ligne les modes "0" et "1" ou "2" et "3" ; les autres combinaisons peuvent donner des résultats inattendus.

Si X contient une valeur autre que 0, 1, 2, 3 ou 9, le calculateur affiche DATA ERROR.

- Arrêt de la fonction "saut de perforations" - SKIPOFF

SKIPOFF annule l'effet de la fonction SKIPON.

- Activation de la fonction "saut de perforations" - SKIPON

SKIPON active la fonction de "saut de perforations". Lorsque cette fonction est active, l'impression de la dernière ligne de texte d'une page (le nombre de lignes de texte dans une page est défini par la fonction TEXTLEN) provoque automatiquement l'avancement du papier jusqu'en haut de la page suivante. On évite ainsi l'impression de texte "à cheval" sur les perforations séparant deux pages.

La fonction "saut de perforations" est inactive à la mise sous tension de l'imprimante ou lors d'une réinitialisation par la fonction CLEAR (voir le mode d'emploi de cette fonction pour plus de détails).

- Longueur de texte - TEXTLEN

TEXTLEN définit le nombre de lignes de texte dans une page .

Le contenu du registre X (sans tenir compte du signe éventuel) indique le nombre de lignes de texte désiré, qui doit être compris entre 1 et le nombre de lignes de la page (défini par la fonction FORMLEN). A la mise sous tension ou lors d'une réinitialisation par la fonction CLEAR (voir le mode d'emploi de cette fonction pour la description de cette possibilité), le nombre de lignes de texte utilisé par l'imprimante est 60.

VSPAC définit l'espacement vertical en nombre de lignes par pouce (1 pouce égale 2,54 cm) selon la valeur présente dans le registre X (le signe étant ignoré).

Cette valeur ne peut être que 6,8,9,12,18,24,36 ou 72 .Si X contient une valeur différente, le calculateur affiche DATA ERROR.

Appendice P

Séquences envoyées à l'appareil principal par les fonctions du groupe 82905 FCNS.

- ESC représente le caractère "échappement", de code décimal 27.
- {#} symbolise la représentation ASCII d'un nombre (paramètre) et {par} les codes des caractères correspondants.

Fonction(s)	Séquence	Codes	Thinkjet
BELL	BEL	07	
CHARSET pour X=0	SI	15	Normal
CHARSET pour X=1 ou -1	SO	14	Gras
FFEED	FF	12	FF
FORMLEN	ESC &l {#} P	27 38 108 {par} 80	FL
GRAPHX	ESC *b {#} G	27 42 98 {par} 71	
MODE	ESC &k {#} S	27 38 107 {par} 83	
			0 Normal (80 c/l)
			1 expansé (40 c/l
			2 compressé (142
			3 expansé-comprés
SKIPOFF	ESC &l0L	27 38 108 48 76	skipoff
SKIPON	ESC &l1L	27 38 108 49 76	skipon
TEXTLEN	ESC &l {#} F	27 38 108 {par} 70	textlen
VSPAC	ESC &l {#} D	27 38 108 {par} 68	vspac

Roman-8 Characters (ASCII)

	CHAR.	DEC.	HEX.	CHAR.	DEC.	HEX.	CHAR.	DEC.	HEX.	CHAR.	DEC.	HEX.
CTL@	NU	0	00		32	20	@	64	40	`	96	60
CTLA	SH	1	01	!	33	21	A	65	41	a	97	61
CTLB	SX	2	02	"	34	22	B	66	42	b	98	62
CTLC	EX	3	03	#	35	23	C	67	43	c	99	63
CTLD	ET	4	04	$	36	24	D	68	44	d	100	64
CTLE	EQ	5	05	%	37	25	E	69	45	e	101	65
CTLF	AK	6	06	&	38	26	F	70	46	f	102	66
CTLG	🔔	7	07	'	39	27	G	71	47	g	103	67
CTLH	BS	8	08	(	40	28	H	72	48	h	104	68
CTLI	HT	9	09	)	41	29	I	73	49	i	105	69
CTLJ	LF	10	0A	*	42	2A	J	74	4A	j	106	6A
CTLK	VT	11	0B	+	43	2B	K	75	4B	k	107	6B
CTLL	FF	12	0C	,	44	2C	L	76	4C	l	108	6C
CTLM	CR	13	0D	–	45	2D	M	77	4D	m	109	6D
CTLN	SO	14	0E	.	46	2E	N	78	4E	n	110	6E
CTLO	SI	15	0F	/	47	2F	O	79	4F	o	111	6F
CTLP	DL	16	10	0	48	30	P	80	50	p	112	70
CTLQ	D1	17	11	1	49	31	Q	81	51	q	113	71
CTLR	D2	18	12	2	50	32	R	82	52	r	114	72
CTLS	D3	19	13	3	51	33	S	83	53	s	115	73
CTLT	D4	20	14	4	52	34	T	84	54	t	116	74
CTLU	NK	21	15	5	53	35	U	85	55	u	117	75
CTLV	SY	22	16	6	54	36	V	86	56	v	118	76
CTLW	EB	23	17	7	55	37	W	87	57	w	119	77
CTLX	CN	24	18	8	56	38	X	88	58	x	120	78
CTLY	EM	25	19	9	57	39	Y	89	59	y	121	79
CTLZ	SB	26	1A	:	58	3A	Z	90	5A	z	122	7A
CTL[	EC	27	1B	;	59	3B	[	91	5B	{	123	7B
CTL\	FS	28	1C	<	60	3C	\	92	5C	|	124	7C
CTL]	GS	29	1D	=	61	3D	]	93	5D	}	125	7D
CTL^	RS	30	1E	>	62	3E	^	94	5E	~	126	7E
CTL_	US	31	1F	?	63	3F	_	95	5F	■	127	7F

MINIPLOTTER

Il existe différents modèles de mini-tables traçantes qui peuvent être avantageusement utilisées avec l'HP-41. TANDY, CANON ... utilisent la même conception mécanique et le même mode de commande . Il est bien entendu nécessaire d'interfacer ces tables tracantes avec la boucle HP-IL au moyen d'un convertisseur GP IO (HP 82166A).

Différentes entreprises mettent sur le marché des tables traçantes interfacées HP-IL, ou proposent un ensemble convertisseur-interface parallèle qui convient parfaitement pour ces tables traçantes.

Les caractéristiques principales de ces tables traçantes sont:

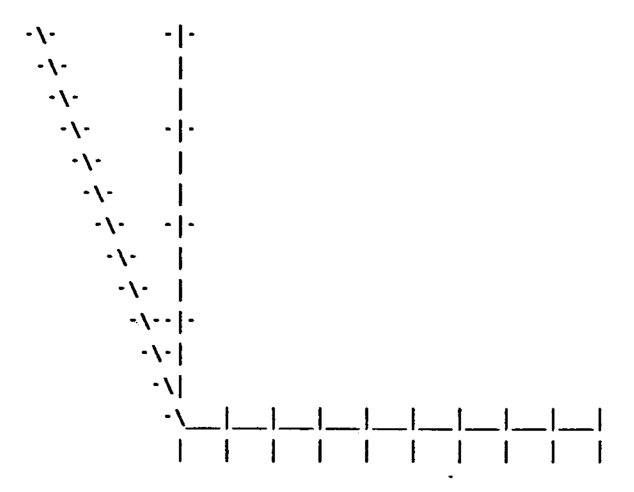
- Quatre couleurs d'impression Noir, Rouge, Bleu, Vert; le tracé est effectué par une 'tête de traçage' qui est en fait un barrillet contenant un jeu de quatre stylos à bille. Le changement de couleur peut être effectué par programme ou par pression d'un interrupteur en face avant au cours du traçage.

- Papier standard de 11,4 cm de large, en rouleau. Il est possible de tracer ou d'écrire en utilisant le papier dans sa longueur, ce qui autorise l'impression de tableau de grande largeur (21 ou 29,7cm). Voyez les exemples.

- Dans la largeur, on peut imprimer des textes de 80 caractères maximum par ligne.

- Bien entendu ces tables traçantes peuvent être contrôlées par d'autres ordinateurs que le HP-41 (HP-75, HP-85, HP-71) et par conséquent ce matériel, comme les autres périphériques HP-IL, ne sera pas "obsolète" trop vite.

AXIS trace toutes sortes d'axes sur le mini-plotter.



INSTRUCTIONS COMPLETES POUR AXIS

AXIS utilise 4 paramètres que l'utilisateur doit placer dans la pile avant l'éxécution de la fonction:
T: demi-longueur de chaque tiret
Z: distance entre 2 tirets dans le sens vertical
Y: distance entre 2 tirets dans le sens horizontal
X: le nombre de tirets

L'axe est tracé à partir de la position courante du stylo, et la direction de tracé dépend uniquement des valeurs placées en Y et en Z. En revanche, les tirets sont toujours soit verticaux soit horizontaux, leur "direction" étant déterminée par l'inclinaison de l'axe par rapport à l'horizontale (direction X): en dessous de 45° les tirets sont verticaux, au-dessus ils sont horizontaux.

Le paramètre situé en T facilite les tracés de grilles. Par exemple pour des tableaux.

Exemple : Le programme suivant trace un tableau à 2 lignes de C colonnes. Chaque colonne ayant une largeur L et chaque ligne ayant une hauteur H. Pour l'utiliser, il suffit de faire **XEQ "TABLO"** et de répondre aux questions posées en introduisant la valeur correspondante suivie de **R/S**.

```
01 LBL "TABLO"
02 "HP82166"        Identification du convertisseur GP-IO.
03 FINDID           Recherche de la position du mini-plotter.
04 SELECT           Selection du mini-plotter.
05 RESET            Réinitialisation.
06 "Nb. COL. ?"
07 PROMPT           Entrée du nombre de colonnes.
08 STO 00           R00= Nombre de colonnes.
09 "LARG. COL. ?"
10 PROMPT           Entrée de la largeur des colonnes.
11 STO 01           R01= Largeur des colonnes.
12 *                1ère dimension du tableau.
13 "HT. LIGNE ?"
14 PROMPT           Entrée de la hauteur de chaque ligne.
15 STO 02           R02= Hauteur de chaque ligne.
16 ST+ X            Il y a 2 lignes, la 2ème dimension vaut 2*X.
17 CHS              Le déplacement aura lieu vers le bas!
18 X<>Y
```

```
0
ENTER^
BOX              BOX utilise les 4 paramètres T, Z, Y et X.
RCL 02
CHS
0
*MOVE            Position de départ.
RCL 02
0
RCL 01
RCL 00
AXIS             Tracé des lignes intérieures.
END
```

```
XEQ "TABLO"
Nb. COL. ?       Imprimera:
 4.0000 RUN
LARG. COL ?
 100.0000 RUN
HT. LIGNE ?
 50.0000 RUN
```

- RECUL D'UN CARACTERE - BACKSP

BACKSP (**BACK SP**ace) déplace le stylo d'un caractère en arrière.

- RECUL DE PLUSIEURS CARACTERES - BACKSPX

BACKSPX (**BACK SP**ace by **X**). Déplace le stylo en arrière du nombre de caractères spécifiés en X. Seule la valeur absolue de la partie entière de X est prise en considération.

- Tracé de boite (rectangle) - BOX

BOX trace un rectangle dont les 2 coins opposés ont pour coordonnées :
(x1,y1) et (x2,y2), avec T= y2, Z= x2, Y= y1 et X= x1.

- Choix de la couleur - *COLOR

*COLOR sélectionne une des quatre couleurs selon la valeur du registre X.

- Taille des caractères - *CSIZE

*CSIZE (Characters SIZE). Selectionne la taille des caractères. La valeur peut être comprise entre 0 et 63.

- Trace un segment - *DRAW

***DRAW** trace un segment de droite à partir de la position courante du stylo jusqu'aux coordonnées (X,Y).

- Positionne le stylo sur l'origine - *HOME

***HOME** ramène le stylo en position (0,0)

- Imprime le contenu du registre ALPHA - *LABEL

***LABEL** imprime le contenu du registre ALPHA. L'utilité de cette instruction tient au fait que l'impression peut être faite dans 4 directions en mode texte; ces 4 directions sont définies par *LDIR.

- RECUL D'UN CARACTERE - BACKSP

BACKSP (**BACK SP**ace) déplace le stylo d'un caractère en arrière.

- RECUL DE PLUSIEURS CARACTERES - BACKSPX

BACKSPX (**BACK SP**ace by **X**). Déplace le stylo en arrière du nombre de caractères spécifiés en X. Seule la valeur absolue de la partie entière de X est prise en considération.

- Tracé de boite (rectangle) - BOX

BOX trace un rectangle dont les 2 coins opposés ont pour coordonnées : (x1,y1) et (x2,y2), avec T= y2, Z= x2, Y= y1 et X= x1.

- Choix de la couleur - *COLOR

*COLOR sélectionne une des quatre couleurs selon la valeur du registre X.

- Taille des caractères - *CSIZE

*CSIZE (Characters SIZE). Selectionne la taille des caractères. La valeur peut être comprise entre 0 et 63.

- Trace un segment - *DRAW

***DRAW** trace un segment de droite à partir de la position courante du stylo jusqu'aux coordonnées (X,Y).

- Positionne le stylo sur l'origine - *HOME

***HOME** ramène le stylo en position (0,0)

- Imprime le contenu du registre ALPHA - *LABEL

***LABEL** imprime le contenu du registre ALPHA. L'utilité de cette instruction tient au fait que l'impression peut être faite dans 4 directions en mode texte; ces 4 directions sont définies par *LDIR.

STATUS place dans le registre Y un nombre entier représentant l'équivalent décimal du premier octet d'état de l'imprimante, et dans le registre X un nombre entier représentant l'équivalent décimal du second octet d'état de l'imprimante. L'effet de STATUS sur la pile dépend de la validation ou non-validation de la montée de la pile au moment de l'exécution de STATUS:

- Si la montée de la pile est autorisée, l'effet est le suivant:

Avant	Après
T: t	T: y
Z: z	Z: x
Y: y	Y: 1er octet d'état
X: x	X: 2ème octet d'état

- Si la montée de la pile n'est pas autorisée, l'effet est le suivant:

Avant	Après
T: t	T: z
Z: z	Z: y
Y: y	Y: 1er octet d'état
X: x	X: 2ème octet d'état

Dans tous les cas, le contenu du registre LASTx n'est pas modifié.

La fonction STATUS possède une caractéristique particulière : en mode MANIO, elle place dans les registres X et Y deux nombres représentant l'état de l'appareil principal.

- Si l'appareil principal ne possède pas d'octet d'état, STATUS place le nombre 97 dans les registres X et Y ;

- Si l'appareil principal possède un seul octet d'état, STATUS place l'équivalent décimal de cet octet dans le registre Y et le nombre 64 dans le registre X ;

- Si l'appareil principal possède au moins deux octets d'état, STATUS fonctionne pour l'appareil principal comme pour l'imprimante HP82162A en mode AUTOIO. Les octets d'état au delà du deuxième sont ignorés.

Pour déterminer le nombre et la définition des octets d'état d'un appareil particulier, se référer à la description du message HPIL "Envoi Etat" ("Send Status") dans le manuel de l'appareil.

L'Appendice S1 donne la définition détaillée des deux octets d'état de l'imprimante HP82162A.

Appendice T2

Jeu minimum de commandes requis pour la gestion complète d'un mini-traceur 4 couleurs par les fonctions du groupe PLOT FCNS.

Pour obtenir une description détaillée d'un mini-traceur se reporter à JPC numéro 15 juin 1984.

Conventions de représentation :

- # représente une chaîne de caractères numériques contenant un signe éventuel et au plus quatre chiffres (exemples : -230 ; 0024) ;

- la colonne "syntaxe" indique la signification de chacun des paramètres.

Caractères de contrôle (valeurs décimales) :

17 : Passage en mode "Texte"
18 : Passage en mode "Graphique"
11 : Retour arrière du papier d'une ligne (mode "Texte" seulement)
08 : Retour arrière du stylo d'un caractère (mode "Texte" seulement)

Instructions en mode "Graphique"

Syntaxe	Format	Action
A	A	Initialisation
H	H	Retour à la position (0,0)
Mx,y	M#,#	Déplacement à la position (x,y)
Dx,y	D#,#	Tracé jusqu'à la position (x,y)
Rx,y	R#,#	Déplacement relatif de (x,y)
Jx,y	J#,#	Tracé relatif de (x,y)
Pchaîne	Pchaîne	Impression de la chaîne de caractères "chaîne"
Lx	L#	Sélectionne le type de ligne x
Cx	C#	Sélectionne le stylo x (changement de couleur)
Sx	S#	Sélectionne la taille de caractères x
Qx	Q#	Sélectionne la direction d'impression x

(pour l'instruction P seulement)

Fonctions de mode "Graphique"

Les instructions du traceur correspondant à ces fonctions nécessitent que le mode "Graphique" soit actif. Ces fonctions placent donc le traceur en mode "Graphique" avant d'effectuer l'opération demandée, et laissent le traceur dans ce mode après exécution.

Fonctions de mode "variable"

Les instructions du traceur correspondant à ces fonctions nécessitent que le mode "Graphique" soit actif. Ces fonctions placent donc le traceur en mode "Graphique" avant d'effectuer l'opération demandée. Néanmoins il est parfois nécessaire d'effectuer ladite opération au cours d'une séquence d'impression utilisant le mode "Texte". L'utilisateur a donc le choix du mode dans lequel le traceur est laissé après l'exécution de ces fonctions.

UTILITAIRES

Les fonctions de ce groupe sont d'utilisation très variée :

- manipulation de chaînes de caractères;

- manipulation de tableaux de nombres ou de chaînes à une ou deux dimensions;

- tri numérique ou alphanumérique;

- gestion de la mémoire étendue (modules XFUNCTIONS HP822180A et XMEMORY HP822181A);

- beaucoup d'autres applications...!

/MOD (Divide MOD) détermine le modulo et le quotient d'une division euclidienne,c'est-à-dire dans l'ensemble des entiers. C'est une extension de la fonction [MOD] du catalogue 3.

Exemple 1 : Calcul du quotient et du modulo de la division de 13 par 3.

Appuyez sur :	Affichage	
13	13_	Introduction du dividende.
[ENTER^] 3	3_	Introduction du diviseur.
[XEQ] "/MOD"	1,0000	Modulo.
[X<>Y]	4,0000	Quotient de la division.
[LAST X]	3,0000	Le diviseur est conservé dans le registre L.

INSTRUCTIONS COMPLETES POUR /MOD

1) Pour calculer le modulo et le quotient de la division de Y par X.
2) [XEQ] "/MOD". Le quotient et le modulo de la division sont replacés respectivement en Y et en X. Le diviseur est conservé dans L, le dividende est perdu. Le contenu des registres T et Z n'est pas altéré.
3) Si le registre X contient zéro, le calculateur affiche DATA ERROR.

La pile

Entrée:	Sortie:
T: T	T: T
Z: Z	Z: Z
Y: dividende	Y: Quotient
X: Diviseur	X: Modulo
L: L	L: Diviseur

PROGRAMMES D'APPLICATION POUR /MOD

Exemple 2 : Voici un procédé assez rapide pour calculer les décimales de la division de A par B lorsque A<B et lorsque B se termine par 9 :

Soit à diviser 153 par 209

```
LBL "DIV9"  10  /  INT  1  +  STO 01  RDN  SF 21
LBL 01  RCL 01  /MOD  VIEW Y  10  *  +  GTO 01
```

153 divisé par 209 = 0,732057...

Exemple 3 : [/MOD] peut être utilisé dans une petite routine comme "changement de base du pauvre" ! Ce court programme, 'YBX' (Y base X), en décomposant le calcul, produit les différents chiffres de la nouvelle valeur à rebours, c'est-à-dire du poids le plus faible au poids le plus fort. X et Y doivent être des entiers.

```
LBL "YBX"  SIGN  LBL 00  X<> L  /MOD  STOP  CLX  X#Y?  GTO 00
BEEP  .END.
```

Par exemple 1103 [ENTER^] 8 [XEQ] "YBX" donne 7 [R/S] 1 [R/S] 1 [R/S] 2 [R/S] 0. Ce qui signifie 1103 (DEC) = 2117 (OCT). Ce résultat peut être vérifié en utilisant les fonctions OCT et DEC.

N.B. : Si l'on veut reconstituer le dividende par X<>Y LASTx * + pour un quotient >0 et par X<>Y X<0? DSE X NOP *LASTx * + pour un quotient #0, on ne peut rien dire pour un quotient nul.

[AD-LC] (ADdress-Line, Column) détermine les coordonnées ligne et colonne d'un élément d'un tableau à partir de son adresse réelle et du pointeur de tableau.

Exemple : Calcul des coordonnées du registre 36 dans le tableau A (ci-dessous) dont le pointeur 25,04405 se trouve dans le registre R00. R25 = N° du premier registre et R44 = N° du dernier registre.

colonne	n°1	n°2	n°3	n°4	n°5
ligne n°1	R25	R26	R27	R28	R29
ligne n°2	R30	R31	R32	R33	R34
ligne n°3	R35	R36	R37	R38	R39
ligne n°4	R40	R41	R42	R43	R44

TABLEAU A

Appuyez sur:	Affichage:	
36 [ENTER^]	36,0000	Introduction du n° de registre.
[RCL] 00	25,04405	Rappel du pointeur de tableau.
[XEQ] "AD-LC"	2,00000	Colonne n°2.
[RDN]	3,00000	Ligne n°3.
[LAST X]	25,04405	Le pointeur est conservé en L.

INSTRUCTIONS COMPLETES POUR AD-LC

Pour calculer les coordonnées ligne et colonne d'un élément d'un tableau, connaissant le pointeur du tableau et le registre occupé par cet élément, introduire le numéro du registre, [ENTER^], introduire le pointeur du tableau, [XEQ] "AD-LC". Le numéro de colonne est retourné en X et celui de la ligne est placé en Y. Le pointeur est conservé en L et les registres Z et T sont inchangés.

La pile

Entrée :	Sortie :
T: t	T: t
Z: z	Z: z
Y: n° de Registre	Y: n° de ligne
X: Pointeur tableau	X: n° de colonne
L: L	L: Pointeur tableau

NOTA: cette fonction ne vérifie pas si le registre fait partie du tableau.
Si les registres X ou Y contiennent une chaîne Alpha, le calculateur délivre le message ALPHA DATA.

[ALENG] (Alpha LENGth) place dans le registre X le nombre de caractères de la chaîne située dans le registre ALPHA.

Exemple 1 : Au cours d'un programme, HP-41 s'arrête pour recevoir une donnée ALPHANUMERIQUE introduite par l'utilisateur. Le programme doit calculer la longueur de cette chaîne afin d'en effectuer le rangement dans différents registres.
N.B. Se reporter à la fonction RGAX dans ce manuel qui fournit une autre solution à ce problème.

INSTRUCTIONS COMPLETES POUR ALENG

Lorsque le registre ALPHA contient la chaîne dont la longueur doit être calculée, [ALENG] place le nombre de caractères dans le registre X et provoque la montée de la pile.

La pile

Entrée:	**Sortie:**
T: t	T: z
Z: z	Z: y
Y: y	Y: x
X: x	X: Longueur de la chaîne ALPHA.
L: l	L: l

PROGRAMMES D'APPLICATION POUR ALENG

Exemple 2 : La routine suivante remplace les minuscules par des majuscules dans la chaîne présente dans le registre ALPHA. Elle utilise [ALENG] pour déterminer le nombre de caratères de la chaîne (à condition que celle-ci ne contienne pas de caractère nul).

```
01 LBL "CAP"
02 ALENG        Détermine le nombre de caractères de la chaîne ALPHA.
03 LBL 00
04 ATOXL        Place en X le code du premier caractère.
05 97           Les codes des minuscules sont compris compris entre 97 et 122.
06 X>Y?
07 GTO 01       Si ce n'est pas une minuscule (<97), aller au LBL 01.
08 CLX
09 122
10 X<Y?
11 GTO 01       Si ce n'est pas une minuscule (>122), aller au LBL 01.
12 CLX          On obtient le code de la lettre majuscule en
13 32           soustrayant 32 du code de la lettre minuscule
14 -            correspondante (A=65, a=97 et 97 - 65 = 32).
15 R^
16 LBL 01
17 RDN
18 XTOAR        Place la lettre majuscule à droite de la chaîne.
19 RDN
20 DSE X
21 GTO 00       Continue la boucle jusqu'à la fin de la chaîne.
22 AON
23 .END.
```

[ANUM] (Alpha to NUMber) : La fonction [ANUM] balaye le contenu du registre ALPHA de gauche à droite à la recherche d'un nombre. Le premier nombre rencontré est placé dans le registre X.

Exemple : Si le registre ALPHA contient la chaîne **PRIX: 1.234,50** obtenue par lecture d'un fichier texte de la mémoire d'extension pour extraire la valeur numérique en vue d'un traitement arithmétique, [XEQ] "ANUM" place ce nombre dans le registre X.

INSTRUCTIONS COMPLETES POUR ANUM

1) La fonction ANUM recherche une valeur numérique dans le contenu de la chaîne ALPHA. Si le calculateur trouve un nombre, il le place dans le registre X et arme l'indicateur 22. Si le calculateur ne trouve pas de nombre, le contenu de X et l'état de l'indicateur 22 ne sont pas modifiés.

2) Les chiffres dans le registre ALPHA peuvent représenter des valeurs dans un format quelconque. Les séparateurs "," et "." sont interprétés conformément à l'état des indicateurs 28 et 29. Si les chiffres dans le registre ALPHA sont précédés d'un signe "moins", le calculateur place un nombre négatif dans le registre X lors de l'exécution de la fonction. En supposant que le registre ALPHA contienne la chaîne de l'exemple n°1:

Indicateur 28	Indicateur 29	Affichage
armé	armé	1,234.5000
armé	désarmé	1,0000
désarmé	armé	1,2345
désarmé	désarmé	1,2340

La pile

Entrée:		**Sortie:**	
T:	t	T:	z
Z:	z	Z:	y
Y:	y	Y:	x
X:	x	X:	valeur trouvée dans ALPHA.
L:	l	L:	l

[ANUMDEL] (Alpha-to-**NUM**ber, **DEL**ete) recherche de gauche à droite dans le registre ALPHA une valeur numérique et place le premier nombre rencontré dans le registre X. Il efface ce nombre en ALPHA ainsi que tous les caractères qui précédent ce nombre.

Exemple 1 : Si le registre ALPHA contient la chaîne **PRIX: 1.234,5 FRS** pour extraire la valeur numérique pour un traitement arithmétique, [XEQ] "ANUMDEL" place ce nombre dans le registre X; la chaîne ALPHA est effacée jusqu'au caractère "5" compris.

INSTRUCTIONS COMPLETES POUR ANUMDEL

1) La fonction ANUMDEL recherche une valeur numérique dans le contenu de la chaîne ALPHA. Si le calculateur trouve un nombre, il le place dans le registre et détruit la chaîne du début jusqu'à ce nombre.

2) Si la chaîne ALPHA contient plusieurs nombres séparés par un ou des caractères non-numériques, [ANUMDEL] ne prend en compte que le premier nombre. [ANUMDEL] est identique à la fonction [ANUM], à ceci près que [ANUM] n'altère pas la chaîne ALPHA. HP-41 considère l'exécution de [ANUMDEL] comme une entrée numérique, et arme l'indicateur binaire F22 quand il place un nombre dans le registre X. Si la chaîne ALPHA ne contient aucune valeur numérique, [ANUMDEL] efface le registre ALPHA mais n'altère pas la pile opérationnelle ni l'indicateur 22.
Les caractères "+" "-" "," "." et "E" (pour l'exposant) sont interprétés par [ANUMDEL] comme des représentations numériques ou non-numériques en fonction du contexte où ils sont placés dans la chaîne. Un "+" isolé, par exemple, n'est pas traité comme un caractère numérique. Un "+" ou un "-" précédant immédiatement, inclus dans, ou suivant directement une séquence de chiffres sera interprété comme une suite de touches introduites au clavier; (avec [CHS] représenté par "-"). Par exemple, [ANUMDEL] renvoie la valeur -3425 si cette fonction est exécutée alors que le registre ALPHA contient la chaîne "34-2+5".

3) Dans le registre ALPHA les chiffres peuvent représenter des valeurs dans un format quelconque. Les séparateurs "," ou "." sont interprétés conformément à l'état des indicateurs 28 et 29. Par exemple, si les indicateurs 28 et 29 sont armés, la virgule est traitée comme un séparateur de tranche de trois chiffres. Mais la virgule sera considérée comme un caractère non-numérique si l'indicateur 28 est armé et l'indicateurs 29 désarmé.
En supposant que la chaîne ALPHA contienne l'exemple n°1: "PRIX: 1,234.5FRS".
Placez-vous en FIX 4

Indicateur 28	Indicateur 29	Nombre affiché	NOUVELLE CHAINE
armé	armé	1,234.5000	FRS
armé	désarmé	1,0000	,234.5FRS
désarmé	armé	1,2345	FRS
désarmé	désarmé	1,2340	.5FRS

La pile

Entrée:	Sortie:
T: t	T: z
Z: z	Z: y
Y: y	Y: x
X: x	X: 1ère valeur trouvée dans ALPHA.
L: l	L: l

PROGRAMMES D'APPLICATION POUR ANUMDEL

Exemple 2 : La table traçante HP 7470A renvoie à la demande une chaîne ASCII décrivant la position courante du stylo. Cette chaîne contient 3 nombres entiers séparés par des virgules: X, Y et P. X est l'ordonnée de la position du stylo, Y l'abscisse, et P prend la valeur 0 ou 1 suivant que le stylo est levé ou baissé. En supposant que la table traçante soit placé en ALPHA "123,456,1", un programme peut traiter ces valeurs en exécutant trois fois de suite [ANUMDEL].

Appuyer sur	Affichage	Commentaires.
[SF 28]		Assure que la virgule ne sera pas considérée comme un séparateur décimal.
[CF 29]		Assure que la virgule ne sera pas considérée comme séparateur de groupe de 3 chiffres.
[ANUMDEL]	123.0000	Ordonnée.
[ANUMDEL]	456.0000	Abscisse.
[ANUMDEL]	1.0000	Le stylo est baissé.

Exemple 3 : ALPHA contient la chaîne "34/-2/5"

[CF 28]	
[ANUMDEL]	34,0000
[ALPHA]	/-2/5
[ALPHA] [ANUMDEL]	-2,0000
[ALPHA]	/5
[ALPHA] [ANUMDEL]	5,0000

Cet exemple met en évidence que les symboles "/" et "*" ne sont pas interprétés comme les symboles "+" "-" ou ".".

[APPX] (**APP**end **X**) copie la partie entière de la valeur présente dans le registre X à droite de la chaîne présente dans le registre ALPHA.

Exemple : Le résultat d'un calcul de surface étant présent dans le registre X : 1.255,7 , et le message d'affichage "SURF: " présent dans le registre, ALPHA [APPX] recopie la valeur présente en X à la suite du message, sans arrondi en valeur entière: ALPHA = "SURF: 1.255"

INSTRUCTIONS COMPLETES POUR APPX

1) [APPX] place la partie entière de la valeur présente dans le registre X à droite de la chaîne ALPHA. [APPX] se conforme à l'état des indicateurs 28 et 29. Le nombre est écrit comme en mode FIX 0 à ceci près que le séparateur de partie décimale n'est pas recopié dans la chaîne, et que le nombre n'est pas arrondi. Comme [ARCL], [APPX] ne provoque pas de signal sonore lorsque l'exécution provoque un dépassement de capacité du registre ALPHA.

Si le registre X contient une chaîne Alpha, le calculateur délivre le message d'erreur ALPHA DATA.

[AROT] (Alpha ROTation) effectue une permutation circulaire du registre ALPHA du nombre de caractères spécifié par le registre X.

Exemple 1 : Le registre ALPHA contenant la chaîne "AROT", si l'on souhaite afficher successivement les messages "TARO" puis "ROTA"

Appuyez sur :	Affichage
[ALPHA] AROT	AROT_
[ALPHA] 1 [CHS]	-1_
[XEQ] "AROT" [ALPHA]	TARO
[ALPHA] 2	2_
[XEQ] "AROT" [ALPHA]	ROTA

INSTRUCTIONS COMPLETES POUR AROT

La fonction [AROT] effectue une permutation circulaire du contenu du registre ALPHA du nombre de caractères spécifié par le registre X modulo 24. La rotation a lieu vers la gauche si le contenu de X est positif et vers la droite si le contenu est négatif. (Cf. l'annexe pour plus d'informations sur l'effet de la fonction [AROT] sur une chaîne contenant des caractères nuls.)

La pile:

L'exécution de la fonction [AROT] ne modifie pas la pile.

PROGRAMMES D'APPLICATION POUR AROT

Exemple 2 : La fonction [AROT] peut être utilisée avec les fonctions [ANUM] et [POSA] pour rechercher les occurences d'un caractère ou d'une chaîne dans ALPHA sans destruction.

Comme résultat d'une opération d'un périphérique, le registre ALPHA contient la séquence 68,2 69,88 (deux nombres séparés par un espace). Supposons qu'il s'agisse d' extraire séparément deux nombres pour les utiliser dans un programme. La séquence suivante illustre la procédure.

Appuyez sur :	Affichage	
[CF] 28		
[XEQ] "ANUM"	68,2000	Place le premier nombre en X.
[STO] 20	68,2000	Conserve pour utilisation.
32	32_	Code de l'espace.
[XEQ] "XTOAR"	32,0000	Ajoute un espace à droite de ALPHA.
[XEQ] "POSA"	4,0000	Recherche le 1er espace dans ALPHA.
[XEQ] "AROT"	4,0000	Effectue une permutation de chaîne; ALPHA contient 69,88 68,2; en l'absence d'espace ALPHA contiendrait 69,8868,2.
[XEQ] "ANUM"	69,8800	Place 69,88 dans X.

- Transfert de caractère entre ALPHA et X -

- Transfert du caractère de gauche de ALPHA dans X - **ATOXL**

[ATOXL] (Alpha-TO-X Left) retire le premier caractère de la chaîne ALPHA et place son code (décimal) dans le registre X.

- Transfert du caractère de droite de ALPHA dans X - **ATOXR**

[ATOXR] (Alpha-TO-X Right) retire le dernier caractère de la chaîne et place son code (décimal) dans le registre X.

- Transfert du caractère spécifié de ALPHA dans X - **ATOXX**

[ATOXX] (Alpha-TO-X by X) place en X le code du caractère spécifié dans le registre X.

INSTRUCTIONS COMPLETES POUR ATOXL, ATOXR et ATOXX

1) [ATOXL] Retire le caractère de gauche de la chaîne contenue dans le registre ALPHA et place son équivalent décimal dans le registre X. Si le premier caractère est suivi d'un ou de plusieurs caractères nuls, ces nuls viennent en tête et la chaîne est décalée vers la gauche jusqu'au premier caractère non nul. [ATOXL] place -1 dans le registre X lorsque le registre ALPHA est vide.

2) [ATOXR] retire le caractère de droite de la chaîne ALPHA et place son équivalent décimal dans le registre X. Si le registre ALPHA est vide, [ATOXR] place la valeur -1 dans le registre X.

3) [ATOXX] recherche le caractère dont la position est spécifiée par le registre X et restitue son équivalent décimal. La chaîne du registre ALPHA n'est pas modifiée.

Une valeur positive de position placée dans le registre X indique une position dans la chaîne ALPHA en comptant de gauche à droite à partir du premier caractère non nul. Ce premier caractère occupe la position 0. Cette convention est identique à celle qu'utilise la fonction [POSA] du module de fonctions d'extension mémoire.

Par contre, une valeur négative placée dans le registre X indique une position absolue dans le registre ALPHA, indépendamment de la chaîne présente dans le registre ALPHA. Les positions sont alors comptées de droite à gauche en commençant par -1 pour le caractère le plus à droite, et en allant jusqu'à -24 pour la position la plus à gauche. Le tableau suivant résume la manière dont [ATOXX] interprète la position des caractères.

Position du caractère	caractère
n > 23 ou n >= longueur de la chaine	Non valable (DATA ERROR)
0 <= n < longueur de la chaine	Nième caractère après le plus à gauche
n = 0	le premier caractère en commençant à gauche
-24 <= n < 0	Nième caractère à partir de la droite et jusqu'à l'extrémité du registre
n < -24	Non valable (DATA ERROR)

Si le registre X contient une chaîne ALPHA, le calculateur délivre le message d'erreur ALPHA DATA.

Exemple :
Dans cet exemple, le registre ALPHA est représenté dans sa totalité, les nuls occupant la partie gauche du registre sont signifiés par les traits horizontaux pour faciliter la compréhension, mais ne peuvent pas être affichés par le calculateur.

Appuyer sur	Affichage	
[ALPHA] DECAMETRE [ALPHA]	——————DECAMETRE	
0 [XEQ] "ATOXX"	68,0000	Code de "D"
4 [XEQ] "ATOXX"	77,0000	Code de "M"
6 [CHS] [XEQ] "ATOXX"	65,0000	Code de "A"
10 [CHS] [XEQ] "ATOXX"	0,0000	Caractère nul

[BLDPT] (**BuiLD** PoinTer) construit un pointeur bbb,eeeii si X est positif ou un pointeur de tableau si X est négatif.

Exemple 1 : Les résultats d'un calcul ont placé dans le registre Z le numéro du premier registre d'un ensemble de valeurs, dans Y le dernier registre, et dans X le nombre de registres séparant chaque valeur consécutive. Z=25 Y=40 X=5

Pour calculer le pointeur, [XEQ] "BLDPT", [FIX] 5
X=25,04005 donnera accès aux registres R25, R30, R35, et R40

Exemple 2 : Les résultats d'un calcul ont placé dans le registre Z le n°du premier registre d'un tableau, dans Y le nombre de lignes et dans X le nombre de colonnes: Z=25 Y=4 X=5.
Pour calculer le pointeur du tableau, [CHS] [XEQ] "BLDPT", X=25,04405

colonne	n°1	n°2	n°3	n°4	n°5
ligne n°1	R25	R26	R27	R28	R29
ligne n°2	R30	R31	R32	R33	R34
ligne n°3	R35	R36	R37	R38	R39
ligne n°4	R40	R41	R42	R43	R44

TABLEAU A

INSTRUCTIONS COMPLETES POUR BLDPT

1) Pour construire un pointeur de la forme bbb,eeeii :

- placer bbb dans le registre Z ;
- placer eee dans le registres Y ;
- placer ii dans le registre X ;
- exécuter [BLDPT] .

2) Pour construire un pointeur de tableau bbb,eeecc où bbb est un nombre représentant le numéro du premier registre occupé par le tableau, où eee est un nombre représentant le numéro du dernier registre occupé par le tableau et où cc est un nombre représentant le nombre de colonnes du tableau :

- placer bbb dans le registre Z ;
- placer le nombre de lignes lll du tableau dans le registre Y ;
- placer le nombre de colonnes cc du tableau dans le registre X, affecté d'un signe négatif ;
- exécuter [BLDPT].

NOTA : Si les registres X,Y ou Z contiennent une chaîne ALPHA, le calculateur délivre le message d'erreur ALPHA DATA.
Le pointeur est construit en prenant les valeurs absolues de bbb et eee.

La pile

pour X>0		Pour X<0	
Entrée:	**Sortie:**	**Entrée:**	**Sortie:**
T: t	T: t	T: t	T: t
Z: bbb	Z: t	Z: bbb	Z: t
Y: eee	Y: t	Y: lll	Y: t
X: ii	X: bbb,eeeii	X: cc	X: bbb,eeecc
L: l	L: eee	L: l	L: lll

[BRKPT] **(BReaK PoinTer)** décompose un pointeur bbb,eeeii si X est positif ou un pointeur de tableau, si X est négatif.

Exemple 1 : Un calcul nécessite de connaître les éléments d'un pointeur bbb,eeeii où bbb est le premier registre d'un ensemble de valeurs, où eee est le dernier registre et où ii est le nombre de registres séparant chaque valeur.
X=25,04005 donne accès aux registres R25, R30, R35 et R40 [XEQ] "BRKPT" restitue Z=25, Y=40 et X=5

Exemple 2 : Le pointeur d'un tableau est 25,04405, indiquant que le tableau commence au registre R25, qu'il s'étend jusqu'au registre R44 et qu'il comporte 5 colonnes. Le nombre de lignes de ce tableau est obtenu par : [CHS] [XEQ] "BRKPT" restitue Z= 25 (premier registre), Y=4 (nombre de lignes), X=-5 (nombre de colonnes)

colonne	n°1	n°2	n°3	n°4	n°5
ligne n°1	R25	R26	R27	R28	R29
ligne n°2	R30	R31	R32	R33	R34
ligne n°3	R35	R36	R37	R38	R39
ligne n°4	R40	R41	R42	R43	R44

TABLEAU A

INSTRUCTIONS COMPLETES POUR BRKPT

1) Pour décomposer un pointeur de la forme bbb,eeeii où bbb est un nombre compris entre 0 et 999 représentant le premier élément d'une boucle ou d'un vecteur, où eee est un nombre compris entre 0 et 999 représentant le dernier élément et où ii est un nombre positif compris entre 0 et 99 représentant le "pas", il convient de s'assurer que la valeur placée dans le registre X est positive, au moyen de [XEQ] [ABS] par exemple, puis [XEQ] "BRKPT" chargera Z avec la partie entière de la valeur qui se trouvait initialement en X, Y avec les trois premiers chiffres de la partie décimale, et X avec les 4ème et 5ème chiffres de cette partie décimale. Le pointeur est conservé dans le registre L.

2) Pour décomposer un pointeur de tableau bbb,eeecc où bbb est le premier registre occupé par le tableau, où eee est le dernier registre et ii le nombre de colonnes, il convient de s'assurer que la valeur placée en X est négative au moyen de [ABS] [XEQ] [CHS] par exemple puis, [XEQ] "BRKPT" retourne séparément le premier registre (bbb) en Z, le nombre de lignes lll=(eee+1-bbb)/cc en Y et le nombre de colonnes (cc) en X.

NOTA : Si le registre X contient une chaîne ALPHA, le calculateur délivre le message d'erreur ALPHA DATA.

La pile

pour X>0		Pour X<0	
Entrée:	**Sortie:**	**Entrée:**	**Sortie:**
T: t	T: x	T: t	T: x
Z: z	Z: bbb	Z: z	Z: bbb
Y: y	Y: eee	Y: y	Y: ll
X: bbb,eeeii	X: ii	X: bbb,eeecc	X: cc
L: l	L: bbb,eeeii	L: l	L: bbb,eeecc où eee=(ll*|cc|)-1+bbb

[CHFLAG] (CHarge FLAGs) place le calculateur au moment de l'exécution du programme dans l'état où se il se trouvait au moment de l'écriture du programme.

Exemple 1 : On souhaite qu'un programme commence par initialiser le calculateur pour qu'il soit en mode DEGré, "ENGénieur" avec 3 chiffres, les 5 premiers indicateurs binaires (0 à 4) armés.
En mode 'RUN' (Indicateur PRGM éteint) configurer le calculateur dans l'état souhaité, puis repasser en mode PRoGraMme [XEQ] "CHFLAG" écrit 2 lignes dans le programme : la première ligne contenant l'instruction CHFLAG, la ligne suivante une chaîne de 7 caractères. A l'exécution du programme, le calculateur se retrouve dans la configuration précédemment définie.

INSTRUCTIONS COMPLETES POUR CHFLAG

1) En mode calcul, placer le calculateur dans l'état souhaité au moment de l'exécution du programme.

2) En mode PRoGraMme, [XEQ] "CHFLAG" écrit deux lignes de programme où la première ligne contient l'instruction CHFLAG, la ligne suivante contient une chaîne de 7 caractères représentant la configuration choisie. Cette chaîne commence par un identificateur qui permet au calculateur de reconnaître qu'il s'agit bien d'une configuration. Si cette chaîne était détruite ou remplacée par une autre non conforme, l'exécution de CHFLAG au cours du déroulement du programme entraînerait l'arrêt du programme et l'affichage du message CHFLAG ERR.

La pile

La pile n'est pas affectée par l'exécution de CHFLAG.

N.B. : Le registre ALPHA n'est pas modifié par l'exécution de [CHFLAG]. La chaîne de caractères est une représentation des indicateurs binaires, elle n'est pas destinée au registre ALPHA.

Il ne faut pas faire précéder 'CHFLAG' d'une instruction de test (telle que ISG ou X=Y?).

```
Ex : FS? 01       Si l'indicateur binaire est armé
CHFLAG            réinitialise le calculateur.
'------'          Chaîne représentant l'initialisation.
```

Si le test est négatif (FLAG 01 clear) provoque la destruction du registre ALPHA par le chargement de la chaîne de représentation.

Seuls les indicateurs binaires 00 à 43 sont pris en compte par "CHFLAG".

F00 à F10 sont les indicateurs réservés à l'utilisateur.

F11 Exécution automatique d'un programme à la mise sous tension ou après une lecture à partir d'une mémoire de masse.

De F12 à F20 Commande de dispositifs externes:

F12 et F13, F15 et F16 sont utilisés par les imprimantes:

F12 double largeur
F13 minuscules

F15	F16	Mode d'impression imprimante HPIL
0	0	Manuel
0	1	Normal
1	0	Trace
1	1	Trace et contenu de la pile

F16
F17 Ignorer CR-LF
F18
F19
F20
F21 Autorisation d'imprimer
F22 Armé par une entrée numérique
F23 Armé par une entrée alpha-numérique
F24 Poursuite malgré erreur de limite
F25 Poursuite malgré erreur
F26 Autorisation de tonalité
F27 Clavier utilisateur
F28 Type de séparateur décimal
F29 Présence ou absence de séparateur de groupes de trois chiffres
F31 Mode DMY du module TIME
F32 Mode MANIO du module HPIL
F34 Mode ADROFF du module Extended I/O
F35 Interdiction démarrage auto (module Autostart / Duplication)
F36 à F39 Nombre de chiffres pour FIX, SCI ou ENG
F40 et 41 Mode d'affichage
F42 et 43 Mode angulaire

[CLINC] (CLear INCrement) tronque la valeur présente dans le registre X à partir du 4ème chiffre de la partie décimale.

Exemple 1 : Pour accéder à toutes les valeurs d'un tableau dont le pointeur 25,04405 est conservé en R00,

Appuyez sur :	Affichage	
[RCL] 00	25,04405	Rappel du pointeur
[XEQ] "CLINC"	25,04400	
[XEQ] "INT"	25,00000	1er élément
[LASTx]	25,04400	
[XEQ] "FRC"	0,04400	
[EEX] 3 [*]	44,00000	Dernier élément

INSTRUCTIONS COMPLETES POUR CLINC

[CLINC] remplace les chiffres décimaux à partir du 4ème inclus de la valeur présente en X, par des 0. L'ancienne valeur est conservée dans le registre L.

La pile

Entrée:	Sortie:
T: t	T: t
Z: z	Z: z
Y: y	Y: y
X: bbb,eeeii	X: bbb,eee
L: l	L: bbb,eeeii

NOTA : Si le registre X contient une chaîne ALPHA, le calculateur délivre un message d'erreur ALPHA DATA.

[COLPT] **(COLumn PoinTer)**, étant donné un numéro de colonne dans le registre Y et un pointeur de tableau dans le registre X, calcule un pointeur de colonne.

Exemple : Afin d'accéder aux registres de la deuxième colonne du tableau A dont le pointeur est conservé dans le registre R00.

Appuyez sur :	Affichage	
2	2_	Numéro de colonne
[RCL] 00	25,04405	Rappel du pointeur
[XEQ] "COLPT"	26,04105	Pointeur de la deuxième colonne

TABLEAU A

colonne	n°1	n°2	n°3	n°4	n°5
ligne n°1	R25	R26	R27	R28	R29
ligne n°2	R30	R31	R32	R33	R34
ligne n°3	R35	R36	R37	R38	R39
ligne n°4	R40	R41	R42	R43	R44

INSTRUCTIONS COMPLETES POUR COLPT

1) Introduire le numéro de colonne dont le pointeur est recherché.

2) Placer dans le registre X le pointeur du tableau auquel appartient la colonne.

3) [XEQ] "COLPT" place en X le pointeur de la colonne et restitue le pointeur du tableau dans le registre L.

La pile

Entrée:	Sortie:
T: t	T: t
Z: z	Z: t
Y: N° de colonne	Y: z
X: bbb,eeeii	X: b'b'b',e'e'e'i'i'
L: l	L: bbb,eeeii

N.B. : i'i'=ii

[GETRGX] (GET ReGisters by X) copie dans les registres spécifiés en X le contenu des registres du fichier courant (fichier dans lequel se trouve le pointeur) à partir de la position du pointeur et en conformité avec l'incrément précisé en X.

Exemple : Le pointeur dans le fichier courant étant en 10, 25,0440510 [XEQ] "GETRGX" copie le contenu des registres 10, 20, 30,... du fichier de la mémoire étendue dans les registres 25, 30, 35,... de la mémoire centrale.

INSTRUCTIONS COMPLETES POUR GETRGX

1) Il convient de s'assurer que le pointeur du fichier courant occupe bien la position souhaitée au moyen de [SEEKPT] ou de [SEEKPTA].

2) La valeur placée dans le registre X est un pointeur de la forme bbb,eeeiijj où bbb est le premier registre de la mémoire centrale dans lequel on souhaite recopier les registres de la mémoire étendue, où eee est le dernier registre où l'on souhaite faire la copie, où ii est l'écart entre 2 registres de la mémoire centrale et enfin, où jj est l'écart entre 2 registres copiés successivement à partir de la mémoire étendue.

3) [XEQ] "GETRGX" copie les registres désignés du fichier courant dans les registres de la mémoire centrale.

La pile
La pile n'est pas modifiée par l'exécution de [GETRGX].

PROGRAMMES D'APPLICATION POUR GETRGX

Les schémas ci-dessous représentent deux tableaux, celui de gauche est placé dans la mémoire principale tandis que celui de droite est placé dans la mémoire d'extension. Dans chaque case sont indiqués le numéro du registre et le contenu du registre (représenté par une lettre).

Commencer par placer le pointeur du fichier sur le premier registre à recopier à partir de la mémoire d'extension en effectuant 12 [SEEKPT].

Pour copier les registres de la 2ème colonne du tableau B de la mémoire étendue dans la 3ème colonne du tableau A de la mémoire principale, il suffit de placer en X le code de la 3ème colonne du tableau A (27,04205) complété du pas des registres à lire dans la mémoire d'extension (03) soit:
X = 27,0420503 .

27 = bbb 1er registre du vecteur en mémoire principale.
42 = eee dernier registre du vecteur en mémoire principale.
05 = ii distance entre 2 registres du vecteur en mémoire principale.
03 = jj distance entre 2 registres lus dans la mémoire étendue.

[XEQ] "GETRGX" recopie les registres conformément au pointeur placé en X; le résultat est présenté dans le deuxième schéma.

PREMIER SCHEMA : DONNEES

Tableau A mémoire principale

col	n°1	n°2	n°3	n°4	n°5
lg 1	R25 A	R26 B	R27 C	R28 D	R29 E
lg 2	R30 F	R31 G	R32 H	R33 I	R34 J
lg 3	R35 K	R36 L	R37 M	R38 N	R39 O
lg 4	R40 Q	R41 R	R42 S	R43 T	R44 U

Tableau B mémoire étendue

col	n°1	n°2	n°3
lg n°1	R11 a	R12 b	R13 c
lg n°2	R14 d	R15 e	R16 f
lg n°3	R17 g	R18 h	R19 i
lg n°4	R20 j	R21 k	R22 l

DEUXIEME SCHEMA : RESULTAT

Tableau A mémoire principale

col	n°1	n°2	n°3	n°4	n°5
lg 1	R25 A	R26 B	R27 b	R28 D	R29 E
lg 2	R30 F	R31 G	R32 e	R33 I	R34 J
lg 3	R35 K	R36 L	R37 h	R38 N	R39 O
lg 4	R40 Q	R41 R	R42 k	R43 T	R44 U

Tableau B mémoire étendue

col	n°1	n°2	n°3
lg n°1	R11 a	R12 b	R13 c
lg n°2	R14 d	R15 e	R16 f
lg n°3	R17 g	R18 h	R19 i
lg n°4	R20 j	R21 k	R22 l

[LC-AD] (Line-Column-ADDress) détermine le numéro de registre d'un élément d'un tableau à partir des coordonnées ligne, colonne et du pointeur de tableau.

Exemple : Détermination du numéro de registre de l'élément placé en ligne 2 et colonne 3 du tableau A dont le pointeur 25,04405 est conservé dans le registre R00.

colonne	n°1	n°2	n°3	n°4	n°5
ligne n°1	R25	R26	R27	R28	R29
ligne n°2	R30	R31	R32	R33	R34
ligne n°3	R35	R36	R37	R38	R39
ligne n°4	R40	R41	R42	R43	R44

TABLEAU A

Appuyez sur:	Affichage:	
[2] [ENTER^]	2,0000	Introduction du n° de ligne
[3]	3_	Introduction du n° de colonne
[RCL] 00	25,04405	Rappel du pointeur de tableau
[XEQ] "LC-AD"	32,00000	N° de registre recherché

INSTRUCTIONS COMPLETES POUR LC-AD

Pour déterminer le numéro de registre d'un élément d'un tableau connaissant le pointeur du tableau, le numéro de ligne et le numéro de colonne: introduire le numéro de ligne, [ENTER^], le numéro de colonne, [ENTER^], le pointeur du tableau. [XEQ] "LC-AD" retourne le numéro de registre en X et conserve le pointeur en L.

La pile

Entrée :	Sortie :
T: T	T: T
Z: n° de ligne	Z: T
Y: n° de colonne	Y: T
X: Pointeur tableau	X: n° de registre
L: L	L: Pointeur tableau

[LINPT] (LINe PoinTer), étant donné le numéro d'une ligne dans le registre Y et un pointeur de tableau dans le registre X, calcule un pointeur de ligne.

Exemple : Afin d'accèder aux registres de la deuxième ligne du tableau A, dont le pointeur est conservé dans le registre R00:

Appuyez sur :	Affichage	
[2]	2_	Numéro de ligne
[RCL] [0] [0]	25,04405	Rappel du pointeur
[XEQ] "LINPT"	30,03400	Pointeur de la 2ème ligne

TABLEAU A

colonne	n°1	n°2	n°3	n°4	n°5
ligne n°1	R25	R26	R27	R28	R29
ligne n°2	R30	R31	R32	R33	R34
ligne n°3	R35	R36	R37	R38	R39
ligne n°4	R40	R41	R42	R43	R44

INSTRUCTIONS COMPLETES POUR LINPT

1) Introduire le numéro de ligne dont le pointeur est recherché.

2) Placer dans le registre X le pointeur du tableau auquel appartient la ligne.

3) [XEQ] "LINPT" place en X le pointeur de la ligne et sauvegarde le pointeur du tableau dans le registre L.

La pile

Entrée:		**Sortie:**	
T:	t	T:	t
Z:	z	Z:	t
Y:	N° de ligne	Y:	z
X:	bbb,eeeii	X:	b'b'b',e'e'e'i'i'
L:	l	L:	bbb,eeeii

[NOP] (No OPeration) est destiné à faire suite à une instruction comportant un test avec branchement conditionnel lorsque le branchement n'est pas utilisé.

Exemple : Dans le cours d'une boucle, on souhaite incrémenter le contenu du registre Y et celui du registre X.

On écrira successivement dans le progamme :

```
ISG Y          Incrémente le registre Y.
NOP            Rend iNOPérant le saut éventuel
ISG X          Incrémente le registre X
GTO 03         et boucle si plus grand
```

[POSA] (POSition in Alpha) balaye de gauche à droite le registre ALPHA à la recherche du caractère ou de la chaîne spécifié dans le registre X.

Exemple 1 : La chaîne "ABCDEFGHIJ" étant placée dans le registre ALPHA, quelle est la position du premier caractère "D" ? Appuyez sur : Affichage 68 68_ Code du caractère D [XEQ] "POSA" 3,0000 Position du caractère D dans le registre ALPHA \ **Exemple 2** : [ALPHA] [CLA] DEF [ASTO] . X ABCDEFGHIJ [ALPHA] [XEQ] "POSA" X=3,00

INSTRUCTIONS COMPLETES POUR POSA

1) [POSA] balaye de gauche à droite le registre ALPHA à la recherche du caractère ou de la chaîne spécifié dans le registre X. La chaîne peut être spécifiée de 2 façons: en entrant le code pour un caractère unique ou en plaçant le caractère ou la chaîne dans le registre X à l'aide de la séquence [ASTO] [.] [X]. Si le calculateur trouve la chaîne dans le registre ALPHA, il place dans le registre X la position de son premier caractère.

2) Les positions sont comptées de gauche à droite en commençant à 0. Si la chaîne ou le caractère apparaît plusieurs fois dans le contenu du registre ALPHA, le calculateur ne donne que la position de la première occurence. Si la chaîne ou le caractère n'existe pas dans le registre ALPHA, le calculateur retourne la valeur -1.

3) La chaîne ou le code du caractère est sauvegardée en L.

La pile

Entrée:		**Sortie:**	
T:	t	T:	t
Z:	z	Z:	z
Y:	y	Y:	y
X:	code ou chaîne	X:	position en ALPHA
L:	l	L:	code ou chaîne

- FONCTIONS D'ALLOCATION DE LA MEMOIRE -

- Programmation de la mémoire - **PSIZE**

[PSIZE] (Programmable **SIZE**) alloue aux données le nombre de registres spécifié par le contenu du registre X.

- Mémoire allouée - **SIZE?**

[SIZE?] place dans X le nombre de registres alloués aux données au moment de son exécution.

Les fonctions [SIZE?] et [PSIZE] peuvent être combinées dans un même programme pour réallouer un ensemble de registres sans détruire des données.

Exemple :

```
01 ....
02 ....
....             Votre programme.
....
07 SIZE?         Le calculateur place dans X le nombre de registres alloués aux données.
08 125           Ce nouveau programme nécessite 125 REG de données.
                 Le résultat précédent est dans le registre Y.
09 X>Y?          Le nombre de registres nécessaires est-il supérieur au nombre actuellement
                                                                                    alloué ?
10 PSIZE         Si oui, réallocation de la mémoire.
```

[READEM] (**READ** Extended Memory) recopie à partir de la mémoire de masse (lecteur de cassette HP82161A par exemple) le contenu des mémoires d'extension préalablement sauvegardé dans ce fichier à l'aide de la fonction WRTEM.

Exemple 1 : Pour charger le fichier "MAT 3" présent sur la cassette

Appuyez sur :	Affichage	
[XEQ] "EMDIR"	DIR EMPTY	Vérifie que la mémoire d'extension est vide. Dans le cas où 2 modules XMEMORY sont présents, le nombre de registres disponibles est alors 600.
[ALPHA] "MAT 3" [ALPHA]	600,0000	ALPHA contient le nom générique des fichiers à lire.
[XEQ] "READEM"		
	600,0000	Les fichiers sont chargés dans la mémoire d'extension.
[XEQ] "EMDIR"	MATRP P012	
	A D100	L'ensemble de ces fichiers
	TEXTE A040	a été lu par READEM.
	....	

INSTRUCTIONS COMPLETES POUR READEM

1) Après avoir placé le nom générique des fichiers à lire [XEQ] "READEM" copie le fichier spécifié de la cassette dans la mémoire étendue.

2) Si le module HPIL n'est pas connecté, le message NO HPIL est affiché.

3) Si le fichier n'existe pas sur la cassette, le message FL NOT FOUND apparaît.

5) Si l'espace de la mémoire étendue est insuffisant, le calculateur affiche NO ROOM. Dans ce cas, ajouter un ou deux modules XMEMORY.

6) Si le module HPIL est connecté et que le lecteur de cassettes est absent de la boucle, le message "NO DRIVE" est affiché et l'exécution est interrompue.

7) Si le fichier n'est pas du type créé par 'WRTEM', 'FLTYPE ERR' est affiché et l'exécution est interrompue.

N.B. [READEM] efface la totalité des fichiers éventuellement situés en mémoire étendue et y place le ou les fichiers préalablement placés dans la mémoire d'extension.

La pile
La pile n'est pas modifiée par [READEM].

FONCTION INVERSE : WRTEM

[RG] est une fonction destinée à faciliter l'entrée au clavier de noms de fonctions commençant par RG. Cette fonction s'utilise essentiellement assignée à une touche. Par exemple, assignée RG à la touche [LN].

ASN "RG" 15

(appuyez sur [ ] [ASN] [ALPHA] [R] [G] [ALPHA] [LN]). Placez le calculateur en mode USER. Dès lors, pour exécuter ou programmer une fonction dont le nom commence par RG, (par exemple RGVIEW), appuyer sur :

[RG] (touche LN) [ALPHA] V I E W [ALPHA]

Cette séquence est équivalente à :

[XEQ] [ALPHA] [R] [G] [V] [I] [E] [W] [ALPHA]

Vous économisez par conséquent 2 pressions de touches à chaque utilisation d'une fonction dont le nom commence par les trois lettres RG.

INSTRUCTIONS COMPLETES POUR RG

1) Assignez [RG] à une touche et placez le calculateur en mode USER.

2) Pour exécuter ou programmer une fonction dont le nom commence par RG, appuyez successivement sur :

[RG] (touche assignée précédemment)
[ALPHA]

.....caractères du nom de la fonction,
.....non compris les 2 premiers
.....(par ex. SUM pour la fonction RGSUM).

[ALPHA]

- OPERATIONS ENTRE LES REGISTRES -

- Somme ou différence terme à terme de 2 vecteurs - **RG+-**

[RG+-] (ReGisters + or -) additionne ou soustrait terme à terme les éléments de 2 vecteurs dont les pointeurs sont spécifiés dans les registres Y et X. C'est le signe de X qui détermine le type de l'opération.

- Multiplication terme à terme de 2 vecteurs - **RG***

[RG*] (ReGisters *) multiplie terme à terme les éléments de 2 vecteurs dont les pointeurs sont spécifiés dans les registres Y et X.

- Divise terme à terme les éléments de 2 vecteurs - **RG/**

[RG/] (ReGisters /) divise terme à terme les éléments de 2 vecteurs dont les pointeurs sont spécifiés dans les registres Y et X.

Exemple : Dans le tableau B ci-dessous:

- remplacer la 1ère colonne par la somme des termes de la 3ème colonne et ceux de la première colonne ;

- puis calculer les carrés de la 4ème colonne ;

- enfin, diviser chacun de ces carrés par les 4 premières valeurs de la première ligne.

Le pointeur du tableau est conservé dans le registre R00.

Tableau avant exécution :
N.B. chaque case contient son numéro de registre et son contenu initial.

colonne	n°1	n°2	n°3	n°4	n°5
ligne n° 1	R25 142	R26 20	R27 857	R28 40	R29 1
ligne n° 2	R30 285	R31 12	R32 714	R33 14	R34 2
ligne n° 3	R35 428	R36 22	R37 571	R38 24	R39 3
ligne n° 4	R40 714	R41 32	R42 285	R43 34	R44 4

TABLEAU B

Appuyez sur :	Affichage	
[CF] 28 [FIX] 5		
[1] [RCL] 00	25,04405	
[COLPT]	25,04005	Pointeur de la première colonne
[3] [RCL] 00	25,04405	
[COLPT]	27,04205	Pointeur de la troisième colonne
[XEQ] "RG+-"	25,04005	Pointeur du vecteur où sont rangés les résultats.

A ce stade, on peut vérifier que les registres R25, R30, R35 et R40 qui constituent la première colonne contiennent tous 999.

```
[4] [RCL] 00          25,04405
[COLPT] [ENTER]       28,04305        X et Y contiennent le pointeur de la 4ième colonne.
[RG] "*"              28,04305
```

A ce stade, les éléments de la 4ème colonne sont :

```
R28= 1600 R33= 196   R38= 576   R43= 1.156
```

```
[1] [RCL] 00          25,04405
[LINPT]               25,02900        Pointeur de la première ligne
[XEQ] "RG/"           28,04305
```

Enfin, la 4ème colonne contient les résultats de la division, et le tableau se présente ainsi :

TABLEAU C

colonne	n°1	n°2	n°3	n°4	n°5
	R25	R26	R27	R28	R29
ligne n°1	999	20	857	1,60	1
	R30	R31	R32	R33	R34
ligne n°2	999	12	714	9,80	2
	R35	R36	R37	R38	R39
ligne n°3	999	22	571	0,67	3
	R40	R41	R42	R43	R44
ligne n°4	999	32	285	722	4

INSTRUCTIONS COMPLETES POUR RG+- RG* RG/

1) Les fonctions [RG+-], [RG*] et [RG/] nécessitent la présence de 2 pointeurs. Le pointeur des opérandes doit se trouver en Y et le pointeur des opérateurs doit être en X.

2) Les résultats sont chargés dans les registres spécifiés par le pointeur placé dans le registre Y.

3) Après l'exécution des calculs, le registre X contient le pointeur où les résultats ont été rangés et le registre L contient le pointeur des opérateurs.

La pile

Entrée:	Sortie:
T: t	T: t
Z: z	Z: t
Y: pointeur n°1	Y: z
X: pointeur N°2	X: pointeur n°1
L: l	L: pointeur n°2

- OPERATION D'UN SCALAIRE SUR DES REGISTRES -

- Addition d'une constante à un ensemble de registres - **RG+Y**

[RG+Y] (ReGisters + Y) additionne la valeur contenue dans le registre Y au contenu des registres spécifiés en X.

- Multiplication d'un ensemble de registres par un nombre - **RG*Y**

[RG*Y] (ReGisters * Y) multiplie le contenu des registres spécifiés en X par la valeur placée en Y.

- Division d'un ensemble de registres par un nombre - **RG/Y**

[RG/Y] (ReGisters / by Y) divise le contenu des registres désignés en X par la valeur placée en Y.

Exemple : Dans le tableau B ci-dessous :

- soustraire la constante 5 aux contenus de la première colonne;
- calculer les doubles des éléments de la 3ème ligne;
- diviser chaque terme de la 5ème colonne par 6.

Le pointeur du tableau est conservé dans le registre R00.

TABLEAU B

colonne	n°1	n°2	n°3	n°4	n°5
	R25	R26	R27	R28	R29
ligne n°1	1	2	3	4	5
	R30	R31	R32	R33	R34
ligne n°1	6	7	8	9	10
	R35	R36	R37	R38	R39
ligne n°1	11	12	13	14	15
	R40	R41	R42	R43	R44
ligne n°1	16	17	18	19	20

Appuyez sur :	Affichage	
[CHS] [ENTER^]	-5,00000	Introduction de la constante.
1 [RCL] 00	25,04405	
[COLPT]	25,04005	Pointeur de la première colonne.
[RG] "+Y"	25,04405	Pointeur du vecteur ou sont rangés les résultats.

A ce stade on peut vérifier que les registres R25, R30, R35 et R40 qui constituent la première colonne contiennent respectivement -4, 1, 6 et 11.

2 [ENTER^]	2,00000	Introduction de la constante.
3 [RCL] 00	25,04405	
[LINPT]	35,03900	X contient le pointeur de la 3ème ligne .
[RG] "*Y"	35,03900	

A ce stade la 3ème ligne contient les doubles des valeurs précédentes R35= 12 R36= 24 R37= 26 R38= 28 R39= 30

6 [ENTER^]	6,00000	Introduction de la constante .

5 [RCL] 00	25,04405	
[COLPT]	29,04405	Pointeur de la 5ème colonne.
[RG] "/Y"	29,04405	

Enfin, la 5ème colonne contient les résultats de la division, et le tableau se présente ainsi:

TABLEAU B

	colonne n°1	n°2	n°3	n°4	n°5
	R25	R26	R27	R28	R29
ligne n°1	-4	2	3	4	0,83
	R30	R31	R32	R33	R34
ligne n°2	1	7	8	9	1,66
	R35	R36	R37	R38	R39
ligne n°3	12	24	26	28	5
	R40	R41	R42	R43	R44
ligne n°4	11	17	18	19	3,33

INSTRUCTIONS COMPLETES POUR RG+Y RG*Y RG/Y

1) Les fonctions [RG+Y], [RG*Y] et [RG/Y] nécessitent la présence d'un pointeur en X et d'une valeur en Y.

2) Les résultats sont chargés dans les registres spécifiés par le pointeur placé dans le registre X, c'est-à-dire qu'ils remplacent les valeurs sur lesquelles sont effectuées les calculs.

La pile

Entrée:	Sortie:
T: t	T: t
Z: z	Z: z
Y: scalaire	Y: scalaire
X: pointeur	X: pointeur
L: l	L: l

La pile est inchangée par [RG+Y], [RG*Y], [RG/Y] .

[RGAX] (**ReGisters-Alpha by X**) 2 fonctions :

1) Si X<0 recopie le registre ALPHA dans les registres spécifiés par le pointeur placé en X,

2) Si X>=0, les registres spécifiés par le pointeur placé en X sont recopiés à la suite de la chaîne présente dans le registre ALPHA.

Exemple : la chaîne ABCDEFGHIJKLMNOPQRSTUVWX est présente dans le registre ALPHA. Pour la conserver dans les registres de n°s pairs à partir de R10, procéder comme suit :

Appuyez sur :	Affichage :	
10,00002 [CHS]	-10,00002_	Pointeur. La valeur négative indique un chargement dans les registres.
[RG] "AX"	-17,00002	Le pointeur désigne le registre suivant la dernière mémoire occupée par le chargement.
[RCL] 10	ABCDEF	6 premiers caractères.
[RCL] 12	GHIJKL	6 caractères suivants.
[RCL] 14	MNOPQR	6 caractères suivants.
[RCL] 16	STUVWX	6 derniers caractères.

On souhaite maintenant replacer dans le registre ALPHA les registres R12 et R16

12,00004	12,00004	Pointeur pour le rappel de la chaîne
[XEQ] "CLA"	12,00004	efface le contenu du registre ALPHA.
[RG] "AX"	17,00004	Pointe le prochain registre.
[ALPHA]	GHIJKLSTUVWX	Notez que le chargement prend fin lorsque le dernier caractère de la chaîne est rencontré.

INSTRUCTIONS COMPLETES POUR RGAX

1) La fonction [RGAX] peut être utilisée pour stocker la totalité du registre ALPHA dans des registres désignés par le pointeur placé en X. Dans ce cas le pointeur doit être une valeur négative. Au moment du chargement, le calculateur place un indicateur de fin de chaîne destiné à la relecture dans le dernier registre utilisé. Cet indicateur n'est pas visible. Cependant, une modification du dernier registre entraîne la perte de ce repère.

2) La fonction [RGAX] peut également être utilisée pour le rappel d'une chaîne conservée dans un ensemble de registres. Dans ce cas le pointeur doit être positif. Le chargement de la chaîne se fait à la suite des caractères déjà présents dans le registre ALPHA. Si la nouvelle chaîne comporte plus de 24 caractères, seuls les 24 derniers resteront en ALPHA. Les premiers caractères de gauche sont perdus. Le chargement s'effectue jusqu'à ce qu'un caractère de fin de chaîne soit trouvé par le calculateur (voir paragraphe précédent), ou, si le calculateur ne trouve pas de repère de fin de chaîne, jusqu'à ce qu'il trouve une valeur numérique dans un registre. Dans ce cas la valeur numérique est chargée dans le format courant de la même façon qu'elle le serait avec [ARCL].

3) Quel que soit le mode d'utilisation, [RGAX] place le pointeur initial dans le registre L et un pointeur ± bbb,eeeii dans le registre X où bbb est le numéro du dernier registre utilisé + 1, et la partie fractionnaire est celle du pointeur initial. D'autre part, les trois premiers chiffres après la virgule peuvent être quelconques, puisque [RGAX] n'en tient pas compte.

La pile

Entrée:		**Sortie:**	
T:	t	T:	t
Z:	z	Z:	z
Y:	y	Y:	y
X:	Pointeur initial	X:	Nouveau pointeur
L:	l	L:	Pointeur initial

[RGCOPY] (ReGisters COPY) possède 2 modes de fonctionnement :

Si **X>=0**, [RGCOPY] copie le contenu des registres désignés par le pointeur placé en X, dans ceux spécifiés par le pointeur placé en Y.

Si **X<0**, [RGCOPY] échange les registres spécifiés en X avec ceux désignés en Y.

Exemple : Dans le tableau B ci-dessous, copier le contenu des registres de la première colonne dans les registres de la 3ème colonne puis échanger les contenus de la 2ème colonne avec ceux de la 1ère ligne.

colonne	n°1	n°2	n°3	n°4	n°5
	R25	R26	R27	R28	R29
ligne n°1	1	2	3	4	5
	R30	R31	R32	R33	R34
ligne n°2	6	7	8	9	10
	R35	R36	R37	R38	R39
ligne n°3	11	12	13	14	15
	R40	R41	R42	R43	R44
ligne n°4	16	17	18	19	20

TABLEAU B

On suppose que le pointeur de ce tableau est contenu dans le registre R00.

Appuyez sur :	Affichage	
3 [RCL] 00 [COLPT]	27,04205	Pointeur de destination.
1 [LAST X] [COLPT]	25,04005	Pointeur d'origine.
[XEQ] "RGCOPY"	27,04205	Pointeur des nouveaux contenus.
[RGVIEW]		liste le contenu de la 3ème colonne R27=1,..., R42=16.
1 [RCL] 00 [LINPT]	25,02900	1er Pointeur.
2 [LAST X] [COLPT] [CHS]	-26,04105	2ème Pointeur.
[RG] "COPY"	25,02900	La pile est descendue.

Le tableau se présente maintenant comme prévu :

colonne	n°1	n°2	n°3	n°4	n°5
	R25	R26	R27	R28	R29
ligne n°1	2	7	12	17	5
	R30	R31	R32	R33	R34
ligne n°2	6	1	6	9	10
	R35	R36	R37	R38	R39
ligne n°3	11	1	11	14	15
	R40	R41	R42	R43	R44
ligne n°4	16	4	16	19	20

TABLEAU B

INSTRUCTIONS COMPLETES POUR RGCOPY

1) Le signe du pointeur placé dans X détermine si les registres sont copiés (X>=0) ou échangés (X<0).

2) La copie s'effectue depuis les registres désignés par le pointeur placé en X dans ceux désignés par le pointeur du registre Y. Après exécution la pile descend.

3) L'échange s'opère entre les registres désignés en X et Y. Après exécution, la pile descend. Si les ensembles de registres ne se chevauchent pas, l'échange s'effectue en commençant par les numéros de registres les plus faibles. S'il y a un chevauchement, le calculateur détermine l'extrémité par laquelle il doit procéder aux échanges afin de ne pas perdre d'information.

La pile

Entrée:	Sortie:
T: t	T: t
Z: z	Z: t
Y: Pointeur destination	Y: z
X: Pointeur d'origine	X: Pointeur destination
L: l	L: Pointeur d'origine

[RGINIT] (ReGisters **INIT**ialize) possède 2 modes de fonctionnement :

Si **X>=0** [RGINIT] place la valeur zéro dans les registres spécifiés par le pointeur placé en X.

Si **X<0** [RGINIT] place les entiers successifs de 1 à N dans les registres spécifiés par le pointeur placé en X.

Exemple : Dans le tableau B ci-dessous dont le pointeur est conservé dans le registre R00, les colonnes 3 et 5 seront remises à zéro puis les colonnes seront numérotées par des nombres de 1 à 5 placés dans la 1ère ligne.

TABLEAU B

colonne	n°1	n°2	n°3	n°4	n°5
	R25	R26	R27	R28	R29
ligne n°1	a	b	c	d	e
	R30	R31	R32	R33	R34
ligne n°2	f	g	h	i	j
	R35	R36	R37	R38	R39
ligne n°3	k	l	m	n	o
	R40	R41	R42	R43	R44
ligne n°4	p	q	r	s	t

Appuyez sur :	Affichage	
3 [RCL] 00 [COLPT]	27,04205	Pointeur de la 3ème colonne.
[XEQ] "RGINIT"	27,04205	Remise à zéro de la 3ème colonne.
5 [LASTx] [COLPT]	29,04405	Pointeur de la 5ème colonne.
[XEQ] "RGINIT"	29,04405	Remise à zéro de la 5ème colonne.
1 [LASTx] [LINPT] [CHS]	-25,02900	Le signe négatif indique une
[XEQ] "RGINIT"	-25,04400	initialisation avec les entiers 1 à N.

TABLEAU B

colonne	n°1	n°2	n°3	n°4	n°5
	R25	R26	R27	R28	R29
ligne n°1	1	2	3	4	5
	R30	R31	R32	R33	R34
ligne n°2	f	g	0	i	0
	R35	R36	R37	R38	R39
ligne n°3	k	l	0	n	0
	R40	R41	R42	R43	R44
ligne n°4	p	q	0	s	0

INSTRUCTIONS COMPLETES POUR RGINIT

1) Lorsque le pointeur placé dans le registre X est positif, les registres spécifiés sont remis à zéro.

2) Lorsque le pointeur placé dans le registre X est négatif, les registres désignés sont chargés avec les entiers successifs de 1 à N.

LA PILE
L'exécution de la fonction [RGINIT] ne modifie pas la Pile.

[RGNb] (ReGisters, NumBer of) calcule le nombre de registres spécifiés par le pointeur placé en X.

Exemple : Pour connaître le nombre d'éléments d'un tableau dont le pointeur est conservé dans le registre R00, puis le nombre de registres contenus dans une ligne:

Appuyez sur :	Affichage :	
[RCL] 00 [CLINC]	25,04400	Pointeur des registres.
[XEQ] "RGNb"	20,00000	Le tableau comporte 20 registes.
1 [RCL] 00 [LINPT]	25,02900	Pointeur de ligne.
[XEQ] "RGNb"	5,00000	Une ligne comporte 5 registres.

INSTRUCTIONS COMPLETES POUR RGNb

[RGNb] place en X le nombre d'éléments désignés par un pointeur bbb,eeeii placé dans le registre X. Le pointeur est conservé dans le registre L.

La pile

Entrée:	Sortie:
T: t	T: t
Z: z	Z: z
Y: y	Y: y
X: Pointeur	X: Nombre d'éléments
L: l	L: Pointeur

[RGSUM] (ReGisters, SUM of) place en X la somme des contenus des registres spécifiés par le pointeur placé en X.

Exemple 1 : Dans le tableau F ci-dessous dont le pointeur est placé en R00, on cherche la somme des élements de la 1ère colonne et la somme des valeurs absolues de la 4ème colonne.

colonne	n°1	n°2	n°3	n°4	n°5
	R25	R26	R27	R28	R29
ligne n°1	-14	15	21	2	8
	R30	R31	R32	R33	R34
ligne n°2	7	13	19	-20	1
	R35	R36	R37	R38	R39
ligne n°3	0	6	12	18	24
	R40	R41	R42	R43	R44
ligne n°4	23	4	5	11	17
	R45	R46	R47	R48	R49
ligne n°5	16	22	3	9	10

TABLEAU F

Appuyez sur :	Affichage :	
1 [RCL] 00 [COLPT]	25,04505	Pointeur de la 1ère colonne.
[RG] "SUM"	32,00000	Somme des éléments.
4 [RCL] 00 [COLPT]	28,04805	Pointeur de la 4ème colonne.
[CHS]	-28,04805	Point. pour valeurs absolues.
[XEQ] "RGSUM"	60,00000	Somme des valeurs absolues.

INSTRUCTIONS COMPLETES POUR RGSUM

[RGSUM] retourne en X la somme des éléments spécifiés par le pointeur placé dans le registre X. Si ce pointeur est négatif, le calculateur effectue la somme des valeurs absolues.

La pile

Entrée:	Sortie:
T: t	T: t
Z: z	Z: z
Y: y	Y: y
X: Pointeur	X: Somme
L: l	L: Pointeur

Exemple 2 : Dans le tableau F ci-dessus, on veut placer dans la 3ème colonne les pourcentages correspondant aux valeurs de la 2ème colonne par rapport à leur somme :

Appuyer sur :	Affichage :	
3 [RCL] 00 [COLPT]	27,04705	Pointeur destination.
2 [LAST X] [COLPT]	26,04605	Pointeur origine.
[RGCOPY]	27,04705	Copie la 2ème colonne à la place de la 3ème.
[XEQ] "RGSUM"	60,00000	Somme des éléments.
[LAST X] [X<>Y]	60,00000	Sauve le pointeur.
100 [/]	0,60000	Prépare le calcul des %.
[X<>Y] [RG/Y]	27,04705	Les calculs sont finis.

Le tableau se présente ainsi :

colonne	n°1	n°2	n°3	n°4	n°5
	R25	R26	R27	R28	R29
ligne n°1	-14	15	25	2	8
	R30	R31	R32	R33	R34
ligne n°2	7	13	21,6	-20	1
	R35	R36	R37	R38	R39
ligne n°3	0	6	10	18	24
	R40	R41	R42	R43	R44
ligne n°4	23	4	6,6	11	17
	R45	R46	R47	R48	R49
ligne n°5	16	22	36,6	9	10

TABLEAU F'

La 3ème colonne contient effectivement les pourcentages correspondant aux valeurs de la 2ème colonne par rapport à leur somme.

[RGVIEW] (ReGisters VIEW) est destiné à différents modes de visualisation et/ou de saisie des registres.

Exemple : Pour effectuer différentes visualisations du tableau ci-dessous : dans certains cas on modifiera les contenus des registres.

colonne	n°1	n°2	n°3	n°4	n°5
	R25	R26	R27	R28	R29
ligne n°1	1	2	3	4	5
	R30	R31	R32	R33	R34
ligne n°2	6	7	8	9	10
	R35	R36	R37	R38	R39
ligne n°3	11	12	13	14	15
	R40	R41	R42	R43	R44
ligne n°4	16	17	18	19	20

TABLEAU I

Appuyez sur :	Affichage	
[CF] 28 [FIX] 6 [<-]	0,000000	
[RCL] 00 [RGVIEW]	25= 1,000000	
	30= 6,000000	Visualise la 1ère colonne.
[R/S]	35= 11,00000	Interruption du catalogue.
[SST]	40= 16,00000	Progression pas à pas.
[BST]	35= 11,00000	Progression à rebours.
[<-]	25,044050	Quitte le catalogue.
[CLINC]	25,044000	Pointeur d'ensemble.
[RGVIEW]	25= 1,000000	
	26= 2,000000	Progression automatique
	27= 3,000000	et visualisation des
	28= 4,000000	registres consécutifs.
[ON]		Eteint le calculateur.
[ON] [CHS]	-25,044000	Pointeur pour arrêt sur la 1ère valeur.
[RGVIEW]	25= 1,000000	
15	25= 15_	Entrée d'une valeur.
[CHS]	25= -15_	
[EEX]	25= -15 _	La saisie est semblable
2 [CHS]	25= -15 -2_	au fonctionnement habituel.
[R/S]	26= 2,000000	Donnée confirmée.
[BST]	25= -0,15000	Vérification.
[SST] [ALPHA]	26= 2,000000	Le mode ALPHA est indiqué.
ABCDEF	26= ABCDEF_	Les données ALPHA sont
G	26= BCDEFG_	acceptées pour 6 caractères.
[<-]	26= BCDEF_	Correction possible.
[R/S] [BST]	26= BCDEF	Donnée validée et vérification.
[SST] A	27= A_	Le mode ALPHA est conservé.
[ALPHA]	27= 3,000000	Retour au mode numérique.
[EEX] 2	27= 1 2_	
[SST] [BST]	27= 3,000000	Donnée non validée par [R/S].
[<-]	-25,044000	Retour au mode calcul.

2 [EEX] 6 [CHS]	2 -6	
[RCL] 00 [+]	25,044052	Pointeur de tableau.
[ALPHA] RIEN A	RIEN A_	Donne un nom au tableau.
[ALPHA] [RGVIEW]	A1,1= -0,150000	Seul le dernier caractère du registre ALPHA sert de nom au tableau.
	A1,2= BCDEF	La progression à lieu automatiquement
	A1,3= 3,000000	à travers la matrice.
[R/S]	A1,4= 4,000000	[R/S] immobilise.
[SST] [SST]	A2,1= 6,000000	Les coordonnées de l'élément
[BST]	A1,5= 5,000000	sont affichées à gauche.
19.5	A1,5= 19,5_	La matrice peut être saisie
[R/S] .	A2,1= ,_	rapidement et clairement.
[R/S]	A2,2= 7,000000	
[BST]	A2,1= 0,000000	
[<--] 6 [EEX] 6 [CHS]	6 -6_	
[RCL] 00 [CHS]	-25,044056	Pointeur de tableau linéaire.
[RGVIEW]	A1= -0,1500000	1er élément de la 1ère col.
[SST]	A2= 0,000000	2ème élément (R30).
[<--] 3 [RCL] 00 [COLPT]	27,042050	Pointeur 3ème colonne.
6 [EEX] 6 [CHS] [+]	27,042056	
[CHS] [RGVIEW]	A1= 3,000000	1er élément (R27).
[ALPHA] LUNDI	A1= LUNDI_	
[R/S] MARDI	A2= MARDI_	Saisie de la colonne
[R/S] MERCR.	A3= MERCR._	élémént par élément.
[R/S] JEUDI [R/S] [ALPHA]	-27,042056	Fin de saisie et retour en mode calcul.
4 [EEX] 6 [CHS] [ENTER]	0,000004	Construction du nouveau
3 [RCL] 00 [COLPT] [+]	27,042054	pointeur.
[CHS] [RGVIEW]	LUNDI=	Dans ce mode [RGVIEW]
29	LUNDI= 29_	accepte des données
[R/S] 12	MARDI= 12_	en conservant à l'affichage
[R/S] [BST]	12,000000=	le contenu précédent.
[<--] 1 [EEX] 6 [CHS]	1 -6	
[RCL] 00 [+] [RGVIEW]	25= -0,150000	Dans ce mode de catalogue
	35= 11,000000	automatique, les valeurs
	40= 16,000000	nulles sont sautées.
	25,044051	

INSTRUCTIONS COMPLETES POUR RGVIEW

1) [RGVIEW] est une fonction générale d'affichage, d'impression et de saisie de données pour les registres de la mémoire centrale.

2) Le pointeur présent dans le registre X détermine l'accès aux registres. Ce pointeur a la forme bbb,eeeiij.

Si **X>=0** : Visualisation successive jusqu'à interruption éventuelle au moyen de la touche [R/S] ou épuisemment des REGistres spécifiés par le pointeur.

Si **X<0** : Visualisation et arrêt sur le 1er REGistre associé à la valeur correspondante. Le registre suivant est obtenu au moyen de la touche [SST]. La touche [R/S] relance le listage de la même façon que lorsque X>=0.

Lorsque j est un nombre impair, les REGistres contenant des valeurs nulles ne sont pas visualisés.

Si **j= 0** ou 1 Le catalogue est normal; c'est à dire que l'affichage présente le n° du registre suivi de son contenu.

Si **j= 2** ou 3 [RGVIEW] affiche les éléments d'un tableau précédés du nom du tableau, du n° de ligne et du n° de colonne.

Si **j= 4** ou 5 [RGVIEW] affiche le contenu du registre suivi du signe "=" . La saisie se fait en conservant l'affichage de ce contenu.

Exemple : Affichage LUNDI=
Saisie LUNDI= 10_

Si j= 6 ou 7 [RGVIEW] affiche le nom du tableau linéaire, le numéro de la case et son contenu.

En mode ALPHA seuls les 6 derniers caractères saisis sont acceptés.

Une imprimante en mode NORMal ou TRACE imprime le catalogue des registres fourni par [RGVIEW].

3) [RGVIEW] se comporte comme un CATalogue (permet BST et SST).

La pile

Entrée:	**Sortie:**
T: t	T: t
Z: z	Z: z
Y: y	Y: y
X: pointeur	X: pointeur
L: l	L: pointeur utilisé

[SORT] (**SORT**er) trie le contenu des registres spécifiés en X.

Exemple : Dans le tableau A ci-dessous :

1) Trier dans l'ordre croissant les valeurs de la 2ème colonne

2) Trier dans l'ordre décroissant les valeurs de la 3ème colonne.

TABLEAU A

colonne	n°1	n°2	n°3	n°4	n°5
	R25	R26	R27	R28	R29
ligne n°1	14	B	21	2	8
	R30	R31	R32	R33	R34
ligne n°2	7	13	19	20	1
	R35	R36	R37	R38	R39
ligne n°3	0	A	-12	18	24
	R40	R41	R42	R43	R44
ligne n°4	23	99	50	11	17

Appuyez sur :	Affichage	
2 [RCL] 00 [COLPT]	26,04105	Construction du pointeur de la 2èME colonne.
[XEQ] "SORT"	SORTING	Tri en cours.
	26,04105	Tri terminé.
3 [LASTx] [COLPT] [CHS]	-27,04205	Pointeur 3ème colonne; le signe négatif indique un tri décroissant.
[XEQ] "SORT"	SORTING	Tri en cours.
	-27,04205	Tri terminé.

TABLEAU A

colonne	n°1	n°2	n°3	n°4	n°5
	R25	R26	R27	R28	R29
ligne n°1	14	13	50	2	8
	R30	R31	R32	R33	R34
ligne n°2	7	99	21	20	1
	R35	R36	R37	R38	R39
ligne n°3	0	A	19	18	24
	R40	R41	R42	R43	R44
ligne n°4	23	B	-12	11	17

INSTRUCTIONS COMPLETES POUR SORT

1) [SORT] trie indifféremment des valeurs numériques ou des chaînes alpha. Les chaînes sont classées suivant le code ASCII et sont considérées comme étant supérieures aux valeurs numériques.

2) Le pointeur placé en X désigne les registres à trier.

3) Si X>=0 les contenus sont classés par ordre croissant.

4) Si X<0 les contenus sont classés par ordre décroissant.

5) Pendant le tri le message "SORTING" est affiché. Si un message est déjà présent à l'affichage, il est maintenu et "SORTING" n'est pas affiché.

La pile
L'exécution de la fonction [SORT] ne modifie pas la Pile.

[STO>L] (STOre by L) Range la valeur présente dans le registre X à l'adresse indiquée par la partie entière du pointeur placé dans le registre L, et incrémente le pointeur placé en L; la montée de la pile n'est pas validée.

Exemple 1 : Pour charger toutes les valeurs de la première ligne d'un tableau de 4 lignes, 5 colonnes et commençant au registre 25

Appuyez sur :	Affichage :	
1	1_	
[RCL] 00	25,04405	Rappel du pointeur
[LINPT]	25,02900	Calcule le pointeur 1ère ligne
[STO] [.] [L]	25,02900	Place le pointeur en L
50	50_	1er élément de la ligne
[XEQ] "STO>L"	50,00000	Range la valeur en R25
[VIEW] [.] [L]	26,02900	Le pointeur a été incrémenté
60	60_	2ème élément de la ligne
[XEQ] "STO>L"	60,0000	Range le 2ème élément
70	70_	
[XEQ] "STO>L"	70,0000	
80	80_	
[XEQ] "STO>L"	80,0000	
90	90_	
[XEQ] "STO>L"	90,0000	
[LASTx]	30,0290	

INSTRUCTIONS COMPLETES POUR STO>L

[STO>L] utilise le contenu du registre L comme pointeur d'adresse pour ranger les valeurs successivement placées dans le registre X.

A chaque éxécution de [STO>L] le contenu de X est transféré dans le registre de données spécifié en L. La montée de la pile n'est pas validée si bien que plusieurs valeurs peuvent être rangées sucessivement sans détruire les registres Y, Z et T. De plus, le pointeur placé en L est automatiquement incrémenté, ce qui économise l'espace mémoire occupé par le programme.

La pile

Entrée:	**Sortie:**
T: t	T: t
Z: z	Z: z
Y: y	Y: y
X: valeur à ranger	X: valeur rangée
L: bbb	L: bbb+1

Remarque : La partie fractionnaire du registre L est ignorée.

NOTA : Si le registre L contient une chaîne ALPHA, le calculateur délivre un message d'erreur ALPHA DATA.

PROGRAMME D'APPLICATION POUR STO>L

Exemple 2 : La fonction [STO>L] est destinée au chargement des valeurs dans des registres en cours de programme. Ainsi pour charger la première colonne du tableau ci-dessous, dont le pointeur se trouve dans le registre R00, la séquence est la suivante :

1 RCL 00 COLPT STO L 50 STO>L 60 STO>L 70 STO>L 80 STO>L

colonne	n°1	n°2	n°3	n°4	n°5
	R25	R26	R27	R28	R29
ligne n° 1	50				
	R30	R31	R32	R33	R34
ligne n° 2	60				
	R35	R36	R37	R38	R39
ligne n° 3	70				
	R40	R41	R42	R43	R44
ligne n° 4	80				

TABLEAU A

[SUB$] (SUBstring) extrait une sous-chaîne du contenu du registre ALPHA ou formate une chaîne par addition d'espaces à droite ou à gauche.

Exemple : Pour extraire 7 caractères à partir de la lettre C de la chaîne 'ABCDEFGHIJKLMNOPQRSTUVW' présente dans le registre ALPHA :

Appuyez sur :	Affichage :	
2,08	2,08	2 est la position du caractère C 8 est la position du 7ème caractère à isoler.
[XEQ] "SUB$"	2,0800	Extrait la sous-chaîne.
[ALPHA]	CDEFGHI	Sous-chaîne.

Pour justifier à droite dans un espace de 10 caractères :

[ALPHA]		
10	10_	Taille de l'espace utilisé.
[CHS]	-10_	Indique la justification à droite.
[XEQ] "SUB$"	-10,0000	
[ALPHA]	CDEFGHI	la chaîne est bien précédée de 3 espaces pour être justifiée.

Pour placer 5 espaces à droite de cette chaîne :

[ALPHA]	-10,0000	
15	15_	Indique le nouvel espace utilisé.
[XEQ] "SUB$"	15,0000	Justification à gauche puisque le registre X est positif.
[ALPHA]	CDEFGHI	
[APPEND]	DEFGHI _	Le registre ALPHA se décale sur la gauche et le curseur attend un nouveau caractère après les 5 espaces.

INSTRUCTIONS COMPLETES POUR SUB$

[SUB$] modifie le contenu du registre ALPHA conformément à l'indication du registre X.

- Si X contient un nombre entier, le calculateur extrait les |X| caractères de droite de la chaîne initiale. Si la chaîne initiale comporte moins de |X| caractères, le calculateur rajoute des espaces pour compléter la chaîne à |X| caractères ; les espaces sont placés à gauche si X est négatif, à droite si X est positif.

- Si X contient un nombre avec partie fractionnaire (bb,ee), le calculateur extrait la sous-chaîne composée des caractères **bb** à **ee** de la chaîne initiale (le caractère le plus à gauche est numéroté 0). Si **ee** est supérieur au numéro du dernier caractère, la sous-chaîne extraite est constituée de la sous-chaîne à partir de **bb** complétée par le nombre d'espaces nécessaires pour obtenir **ee-bb+1** caractères. Les espaces rajoutés le sont à droite si X est positif, à gauche sinon. Attention le signe de X n'intervient pas si **ee** est inférieur ou égal au numéro du dernier caractère de la chaîne initiale.

Si **bb** est supérieur au numéro du dernier caractère de la chaîne initiale, SUB$ place dans le registre ALPHA une chaîne constituée de **ee-bb+1** espaces.

La pile

La pile n'est pas altérée par l'exécution de [SUB$].

NOTA : Si le registre ALPHA contient 24 caractères le calculateur place des caractères de code zéro en tête de chaîne qui apparaîtront sous la forme de petits traits précédant la chaîne.

[TF55] (Toggle **Flag 55**) inverse l'état de l'indicateur binaire 55, qui indique normalement si une imprimante est connectée au HP41. Cet indicateur n'est pas manipulable par l'utilisateur ne disposant pas du module PANAME. L'ordre TF55 :

1) Arme l'indicateur binaire 55 lorsqu'il n'y a pas d'imprimante connectée au HP41; cela facilite l'utilisation de certains programmes (disponibles par exemple dans des modules d'application), qui sont exécutés obligatoirement avec l'indicateur binaire 21 ("Validation d'imprimante") armé, d'être utilisés comme sous-programes. En effet, de tels programmes s'interrompent lorsqu'ils rencontrent les instructions VIEW et AVIEW si l'indicateur binaire 55 est désarmé. TF55, parce qu'il arme cet indicateur, évite ces interruptions.

2) Désarme l'indicateur binaire 55 lorsqu'une imprimante est connectée au HP41, ce qui accélère notablement l'exécution des programmes tant que l'imprimante n'a pas à intervenir; l'imprimante est rétablie par un nouveau TF55.

INSTRUCTIONS COMPLETES POUR TF55

1) Pour armer l'indicateur binaire 55 alors qu'il est désarmé, exécuter TF55;

2) Pour désarmer l'indicateur binaire 55 alors qu'il est armé, exécuter TF55.

[VKEYS] (View KEYS) fait défiler à l'affichage du HP41 la liste des assignations de touches (redéfinitions accéssibles en mode USER), en progressant à partir de la touche R/S, de haut en bas et de droite à gauche. A titre d'illustration, si la fonction PROMPT est assignée à la touche "ENG" (touche jaune suivie de [3], code de la touche -74), le calculateur affichera :

-74 PROMPT

le listage des assignations peut être :

- temporairement interrompu par la pression maintenue d'une touche autre que R/S ou ON;

- définitivement interrompu par la pression de la touche R/S ou de la touche ON qui, en outre, met le calculateur hors tension.

Note : la fonction VKEYS est non programmable.

[WRTEM] (Write Extended Memory) recopie la mémoire d'extention sur un support (cassette HP82161A ou disquette HP9114).

Exemple 1 : Pour charger le fichier "MAT 3" sur la cassette

Appuyez sur :	Affichage		
[XEQ] "EMDIR"	MATRP	P012	
	A	D100	L'ensemble de ces fichiers
	TEXTE	A040	a été lu par READEM.
	....		
[ALPHA] "MAT 3" [ALPHA]	600,0000		ALPHA contient le nom générique des fichiers à lire.
[XEQ] "WRTEM"			
	600,0000		Les fichiers sont chargés sur la cassette.

INSTRUCTIONS COMPLETES POUR WRTEM

1) Après avoir placé le nom générique des fichiers à lire [XEQ] "WRTEM" copie le fichier spécifié sur la cassette de la mémoire étendue.

2) Si le module HPIL n'est pas connecté, le message NO HPIL est affiché.

3) Si le fichier existe sur la cassette, le fichier est remplacé. **La pile** \La pile n'est pas modifiée par [WRTEM].

FONCTION INVERSE : WRTEM

La fonction **X<>F** échange les contenus du registre X et d'un registre fictif F qui contient une représentation de l'état des indicateurs binaires 00 à 07. Cette représentation est un nombre entier compris entre 0 et 255, somme des valeurs correspondant aux indicateurs binaires armés :

Indicateur	Valeur
0	1
1	2
2	4
3	8
4	16
5	32
6	64
7	128

Par exemple, si les indicateurs 0, 1 et 3 sont armés, les indicateurs 2, 4, 5, 6 et 7 étant désarmés, le registre "F" contient :

1 (valeur de l'indicateur 0)
+2 (valeur de l'indicateur 1)
+8 (valeur de l'indicateur 3)
=11

INSTRUCTIONS COMPLETES POUR X<>F

Pour définir un nouvel état des indicateurs binaires 00 à 07, et simultatément obtenir la représentation de leur état actuel :

1) Calculer (cf. ci-dessus) la représentation R' du nouvel état désiré, et placer cette valeur dans le registre X;

2) Exécuter X<>F.
Dès lors, le registre X contient R, représentation de l'état des indicateurs binaires 00 à 07 avant l'éxécution de X<>F, et le nouvel état des indicateurs correspond à la représentation R'.

Exemple d'application de X<>F :

Le programme XFLAGS suivant permet d'utiliser (jusqu'à) 80 indicateurs binaires d'usage général. Ces "indicateurs binaires étendus" (I.B.E) numérotés de 0 à 79, s'utilisent comme suit :

- pour armer l'I.B.E. numéro N, placer N dans le registre X et exécuter XSF.

- pour désarmer l'I.B.E. numéro N, placer N dans le registre X et exécuter XCF.

- pour tester l'I.B.E. numéro N, placer N dans le registre X et exécuter XFS?.
dès lors, l'indicateur binaire 08 du HP41 possède le même état (armé ou désarmé) que l'I.B.E. numéro N.

Les programmes XSF, XCF et XFS? utilisent la pile, les registres R00 à R09; XFS? utilise en outre l'indicateur binaire 08.

Listing du programme XFLAGS :

```
LBL "XFLAGS"  .009  RGINIT  RDN  RTN
LBL "XSF"  XEQ 00  SF IND Y  GTO 01
LBL "XCF"  XEQ 00  CF IND Y  GTO 01
LBL "XFS?"  XEQ 00  CF 08  FS? IND Y  SF 08
LBL 01  X<>F  STO IND Z  R^  RTN
LBL 00  STO Y  8  /MOD  RCL IND Y  X<>F  .END.
```

Note : XEQ "XFLAGS" désarme tous les I.B.E. 00 à 79

Les fonctions X#NN?, X<=NN?, X<NN?, X<=NN?, X>=NN? et X>NN? sont similaires aux fonctions de comparaison standard (ex: X=Y?) du HP41, mais comparent non pas les contenus des registres X et Y, mais le contenu du registre X avec le contenu du registre spécifié par le registre Y.

Ces fonctions, en outre comparent des chaînes ALPHANUMERIQUES.

INSTRUCTIONS COMPLETES POUR X...NN?

Pour effectuer une comparaison entre le contenu du registre X et le contenu d'un registre r, placer dans Y la valeur suivante :

Si le registre r est :	Placer dans Y :
- un registre de données Rnnn	- Le nombre nnn
- le registre Z	- la chaîne 'Z' ('Z' ASTO.Y)
- le registre T	- la chaîne 'T'
- le registre L	- la chaîne 'L'

puis effectuer la comparaison désirée.

En mode calcul, le HP41 affiche YES ou NO suivant le résultat de la comparaison.

Au cours de l'éxécution d'un programme, la ligne de programme placée derrière le test est exécutée si le résultat de la comparaison est OUI, elle est ignorée dans le cas contraire, comme pour toutes les opérations de test habituelles du HP41.

Ces fonctions comparent des nombres et des chaînes alphanumériques, avec les conventions suivantes :

1) un nombre est toujours strictement inférieur à une chaîne.

2) les chaînes sont ordonnées à partir des codes de leurs caractères
(ex : 'AB0' < 'ABA' car le code de '0' est 48 et celui de 'A' 65.

3) une chaîne courte identique au début d'une chaîne plus longue est considérée comme inférieure. (ex : 'ABC' < 'ABCD').

[Y/N] simplifie les programmes qui, lors de leur utilisation, posent à l'utilisateur une question à laquelle il faut répondre par OUI ou par NON.

Exemple : La séquence de programme suivante fait afficher au calculateur la question :

FIN Y/N?

et dirige l'exécution du programme vers le label 00 si l'utilisateur répond OUI à la question (par la pression de la touche O) ou au label 01 si l'utilisateur répond NON à la question (par la pression de la touche N) :

```
"FIN"       1,0000          Reste de la division.
Y/N
GTO 00
GTO 01
```

INSTRUCTIONS COMPLETES POUR Y/N

La fonction Y/N ne s'utilise qu'à l'intérieur d'un programme.

1a) Pour poser une question de la forme :

message Y/N?

placer le "message" (max. 7 caractères) dans le registre ALPHA et exécuter Y/N;

1b) Pour poser une question sous une autre forme (par exemple : FIN O/N) placer le message dans le registre ALPHA, exécuter AVIEW puis Y/N.

2) Dans tous les cas, lors de l'exécution de Y/N le calculateur s'arrête et attend la pression d'une touche :

- si la touche pressée est ON, le calculateur est mis hors tension;

- si la touche pressée est R/S, l'exécution du programme est suspendue et le pointeur programme est placé sur la ligne suivant immédiatement Y/N;

- si la touche pressée est Y (Yes) ou O (Oui) l'exécution du programme se poursuit à la ligne suivant immédiatement Y/N;

- si la touche pressée est N (No ou Non), la ligne suivant immédiatement Y/N est ignorée, et l'exécution du programme se poursuit à la 2ème ligne suivant Y/N (comme dans le cas d'un test de résultat FAUX - voir par ex. le mode d'emploi de la fonction X=Y? dans le manuel d'utilisation du HP41).

- toute autre touche est ignorée.

Appendice ON

Cet appendice décrit les "fonctions" supplémentaires dont dispose le calculateur HP41 lors de la mise sous tension par la touche [ON]. Les "fonctions" sont similaires à la fonction de réinitialisation du calculateur, qui est obtenue par pression simultanée des touches [ON] et [<-] à la mise sous tension.

Notation : ON/+ symbolise la "fonction" obtenue par maintien de la touche + enfoncée en même temps que la touche ON lors de la mise sous tension du calculateur, la touche + étant relâchée après la touche ON.

ON/.

Fait passer du mode d'affichage "américain" des nombres (1.2345) au format "européen" (1,2345) et vice-versa. Cette "fonction" existe sur les calculateurs de la série 10 (HP10C, HP11C, HP12C, HP15C et HP16C). Notez que cette fonction inverse l'état de l'indicateur binaire numéro 28.

ON/K

Supprime toutes les assignations de touches obtenues par la fonction ASN et actives en mode USER.

ON/A

Effectue les assignations du "jeu A" décrit ci-après. Si une des touches utilisées par ces assignations comporte déjà une assignation, celle-ci n'est pas modifiée.

ATOXL	ALENG	ATOXX	ANUMDEL	ATOXR
XTOAL	AROT	YTOAX	ANUM	XTOAR

ON/M

Comme ON/A, mais active le jeu d'assignations M décrit ci-après.

STO>L	BRKPT	COLPT	AD-LC	RGVIEW
RG	BLDPT	LINPT	LC-AD	CLINC

ON/T

Comme ON/A, mais active le jeu d'assignations T décrit ci-après.

AXIS	BOX	SETORG	RMOVE	*CSIZE
*HOME	RESET	*LABEL	*MOVE	*LDIR
*PLREGX	REVLFX	BACKSPX	RDRAW	*LTYPE
		OUT	*DRAW	COLOR

ON/V

Comme ON/A, mais active le jeu d'assignations V décrit ci-après.

	SCRLUP	CLEAR	XYTAB	CTYPE
HOME	SCRLDN	CLEARO	CSRL	CSRR
	SCRLX	CSRVX	CSRUP	CSROFF
		CSRHX	CSRDN	CSRON

ANNEXE

Le HP-41 utilise la notation polonaise inverse pour résoudre des problèmes complexes, sans parenthèses et avec un minimum de pressions de touches. Ce système a été "inventé" par un célèbre mathèmaticien polonais nommé Lukasiewicz et non par la société Hewlett-Pacckard. S'il est certain que la maîtrise parfaite de ce système peut prendre quelques heures pour un utilisateur habitué à d'autres écritures, il est encore plus certain que le gain de temps et de patience consécutif le rembourse très largement de cet effort :

- UNE ECONOMIE DE TEMPS : l'accès aux registres mémoire est, sur ce type de calculateurs, beaucoup plus rapide dans la pile quelle que soit la partition choisie; (SIZE)

- UNE ECONOMIE D'ESPACE : un résultat intermédiaire qui n'occupe pas un registre le laisse disponible pour autre chose ;

- mais surtout travailler dans la pile, c'est éviter d'accroître la COMPLEXITE DE LA GESTION DE LA MEMOIRE, et pouvoir utiliser une procédure sans avoir à la modifier pour tenir compte de l'implantation des variables du programme appelant.

Ainsi, les paramètres d'une procédure sont transmis par la pile, les calculs effectués dans la pile, les résultats finalement récupérés à partir de la pile : la procédure forme alors un tout insécable, transposable, universel.

En règle générale, tout traitement arithmétique de quatre valeurs peut toujours être traité dans la pile.

Exemple 1 : Calcul des racines de l'équation du second degré

ax^2 + bx + c = 0

Pour exécuter le programme ci-dessous, introduire successivement :

c ENTER^ b ENTER^ a XEQ "RACINE" .

```
LBL "RACINE" ST/ Z CHS ST+ X / ENTER^ X^2
RCL Z - SQRT RCL Y SIGN * + ST/ Y .END.
```

Essayer par exemple de résoudre les équations

x^2 + x -6 = 0 et

3x^2 + 2x -1 =0

-6 ENTER^ 1 ENTER^ XEQ "RACINE" . x'= -3 et RDN x"= 2

Pour l'autre essai -1 ENTER^ 2 ENTER^ 3 XEQ "RACINE"
x'= -1 et RDN x"= 0,3333

Cet exemple montre la souplesse que confère l'arithmétique directe dans la pile.

Exemple 2 : calcul du PGCD de 2 nombres

```
01 LBL "PGCD" LBL 02 MOD LASTX X<>Y X#0? GTO 02 + .END.
```

91 ENTER^ 65 XEQ "PGCD" . X = 13,00

L'essai précédent montre que pour utiliser ce court progamme, il convient d'introduire le couple de valeurs en Y et en X, dans un ordre quelconque; l'appel de "PGCD" renvoie en X le résultat.

Exemple 3 : Calcul de fraction réduite

Le calcul du PGCD est habituellement destiné à obtenir la forme réduite d'une fraction (simplification des fractions). Disposant d'une procédure de calcul du PGCD, il est aisé de calculer la fraction réduite, le PGCD et le PPCM des termes de la fraction.

```
01 LBL "RF" STO Z X<>Y STO T XEQ "PGCD" ST/ Z ST/ Y RDN ST* Z .END.
```

Essai : 91 ENTER^ 65 XEQ "RF" fournit X= 5 et Y=7 c'est à dire 91 / 65 = 7 / 5 ; le PPCM et le PGCD de 91 et 65 sont Z = 455 et T = 13

Exemple 4 : Calculs effectués entre 2 fractions.

La régle générale, énoncée plus haut, trouve ici pleinement son application, puisqu'il s'agit d'effectuer des calculs sur 4 valeurs.

```
LBL "F/" X<>Y LBL "F*" ST* Z RDN ST* Z RDN GTO "RF"
09 LBL "F-" CHS LBL "F+" ST* T X<> Z ST* Z * RCL Z + X<>Y GTO "RF" .END.
```

Essai : quelle résistance mettre en parallèle avec une autre de 100 ohms pour obtenir une valeur finale de 80 ohms ?

1 ENTER^ 80 ENTER^ 1 ENTER^ 100 XEQ "F-" . La valeur recherchée est 400

A travers ces exemples, il est possible de se rendre compte de la puissance de la pile opérationnelle.